Ελληνικό Database-SQL-RDBMS HOW-TO για Linux (PostgreSQL -Αντικειμενοστραφές Σχεσιακό Σύστημα Βάσης Δεδομένων)

Al Dev (Alavoor Vasudevan) alavoor@yahoo.com v11.0, 8 Ιανουαρίου 1999

Αυτό το κείμενο είναι ένας "πρακτικός οδηγός" για να εγκαταστήσετε γρήγορα την SQL Database engine και τα front end εργαλεία της σε ένα σύστημα Unix. Μιλά επίσης για τη Διεθνή γλώσσα Προτύπων (International Standard language) ANSI/ISO SQL και εξετάζει τα πλεονεκτήματα της SQL database engine που έχει αναπτυχθεί από το world-wide internet κάτω από ένα "ανοικτό περιβάλλον ανάπτυξης". Πρόκειται για το ΠΩΣ (HOW-TO) να εγκαταστήσετε την Αντικειμενοστραφή Σχεσιακή SQL Βάση επόμενης γενιάς "PostgreSQL" σε ένα σύστημα Unix το οποίο μπορεί να χρησιμοποιηθεί σαν Application Database Server ή σαν Web Database Server. Η PostgreSQL υλοποιεί ένα υποσύνολο του Διεθνούς Προτύπου ISO και ANSI SQL 1998,92,89. Αυτό το κείμενο δίνει επίσης πληροφορίες για τα προγράμματα διασύνδεσης με τη βάση όπως τα Front End GUIs, για εργαλεία RAD (Rapid Application Development), για interfaces γλωσσών προγραμματισμού "C", "C++", Java, Perl, ODBC, JDBC οδηγούς και εργαλεία σύνδεσης της βάσης με Web και προγράμματα γραφικής διασύνδεσης. Οι πληροφορίες που δίνονται εδώ ισχύουν για όλες τις πλατφόρμες Unix και για άλλες βάσεις επίσης και θα είναι πολύ χρήσιμες σε άτομα νέα στο χώρο των Βάσεων, της γλώσσας SQL και της PostgreSQL. Αυτό το κείμενο έχει επίσης οδηγό εκμάθησης της SQL, και η σύνταξη SQL θα φανεί πολύ χρήσιμη στους αρχάριους. Οι έμπειροι θα βρουν χρήσιμο αυτό το κείμενο σαν πρόχειρο οδηγό αναφοράς. Για τους φοιτητές, οι πληροφορίες που δίνονται εδώ θα τους κάνει ικανούς να πάρουν τον πηγαίο κώδικα από τη σχεσιακή βάση δεδομένων PostgreSQL, απ' όπου μπορούν να μάθουν πώς γράφεται μία σχεσιακή SQL βάση δεδομένων.

Contents

1 Εισαγωγή

Ο σκοπός αυτού του κειμένου είναι να δώσει κατανοητούς δείκτες/URLs για να στήσετε γρήγορα την PostgreSQL και επίσης να υπερασπιστεί τα πλεονεκτήματα ενός Ανοικτού Συστήματος Πηγαίου Κώδικα (Open Source Code) όπως η PostgreSQL, Linux.

Κάθε σύστημα υπολογιστή στον κόσμο χρειάζεται μια βάση δεδομένων για να αποθηκεύει /ανακτά πληροφορίες. Ο κύριος λόγος που χρησιμοποιούμε τον υπολογιστή είναι για να αποθηκεύουμε, να ανακτούμε και να επεξεργαζόμαστε την πληροφορία και να τα κάνουμε όλα αυτά γρήγορα, εξοικονομώντας έτσι χρόνο. Ταυτόχρονα, το σύστημα πρέπει να είναι απλό, αξιόπιστο, οικονομικό και πολύ εύκολο στη χρήση του. Η Βάση Δεδομένων είναι **ΣΥΣΤΗΜΑ ΖΩΤΙΚΗΣ ΣΗΜΑΣΙΑΣ** για κάθε εταιρία στον κόσμο μια και σε αυτή αποθηκεύονται πολύ σημαντικές πληροφορίες. Τα πιο δημοφιλή συστήματα βάσεων δεδομένων βασίζονται στις SQL προδιαγραφές του Διεθνούς Οργανισμού Προτύπων ISO (International Standard Organisation) οι οποίες βασίζονται με τη σειρά τους στ ο πρότυπο ANSI SQL (Αμερικάνικο). Οι τρέχουσες προδιαγραφές που χρησιμοποιούνται ευρέως είναι οι ISO/ANSI SQL 1992. Το μελλοντικό πρότυπο είναι το SQL 1998/99 γνωστό και σαν SQL-3 και αναπτύσσεται ακόμη. Δημοφιλείς βάσεις όπως η Oracle, η Sybase και Informix είναι συστήματα βασισμένα σε αυτά τα πρότυπα ή προσπαθούν να τα υλοποιήσουν.

Όπως φαίνεται σε αυτό το κείμενο, υπάρχουν περισσότερα από 20 διαφορετικά συστήματα βάσεων δεδομένων, εμπορικά/βασισμένα στο internet, που χρησιμοποιούνται σε όλο τον κόσμο και πολλά περισσότερα θα εμφανιστούν στο προσεχές μέλλον. Χωρίς ένα πρότυπο όπως το ANSI/ISO SQL, θα ήταν πολύ δύσκολο για τον πελάτη να αναπτύξει μια εφαρμογή μια μόνο φορά και να τη τρέχει σε όλα τα συστήματα βάσεων δεδομένων. Ο σημερινός πελάτης θέλει να αναπτύξει μια εφαρμογή μια φορά χρησιμοποιώντας ISO SQL, ODBC, JDBC και να την πουλήσει ώστε να χρησιμοποιείται σε όλα τα διαφορετικά συστήματα βάσεων δεδομένων στον κόσμο.

Η δημοφιλέστερη παγκοσμίως ΔΩΡΕΑΝ βάση, που υλοποιεί τμήμα του ISO SQL, ANSI SQL/98, SQL/92 και ANSI SQL/89 RDBMS είναι η PostgreSQL. Η PostgreSQL είναι η Αντικειμενοστραφής Σχεσιακή SQL Βάση επόμενης γενιάς και τα μελλοντικά πρότυπα ANSI SQL όπως το SQL 1998 (SQL-3) και πέρα θα ασχολούνται όλο και περισσότερο με Αντικειμενοστραφείς Βάσεις και τύπους δεδομένων Αντικειμένου (Object data types). Η PostgreSQL είναι το μόνο δωρεάν RDBMS στον κόσμο που υποστηρίζει Αντικειμενοστραφείς Βάσεις και SQL. Αυτό το κείμενο θα σας πει πώς να εγκαταστήσετε τη βάση και, επίσης, όλα τα πακέτα σχετικά με βάσεις, πώς να εγκαταστήσετε Web database, application database, γραφικά προγράμματα επικοινωνίας και χειρισμού (front end GUIs και προγράμματα interface). Σας συμβουλεύουμε να γράφετε ΟΠΩΣΔΗΠΟΤΕ τις εφαρμογές βάσεων 100 % συμβατές με τα πρότυπα ISO/ANSI SQL, ODBC, JDBC, έτσι ώστε η εφαρμογή σας να είναι μεταφέρσιμη σε πολλές βάσεις όπως οι PostgreSQL, Oracle, Sybase, Informix κλπ.

Με την PostgreSQL, θα έχετε την υψηλότερη ποιότητα και πολλά άλλα χαρακτηριστικά μια και ακολουθεί το 'Ανοικτό Μοντέλο Ανάπτυξης Πηγαίου Κώδικα' (Open Source Code development). Το Ανοικτό μοντέλο Ανάπτυξης Πηγαίου Κώδικα είναι το μόνο του οποίου μπορείτε να πάρετε όλον τον πηγαίο κώδικα, και η ανάπτυξη γίνεται στο internet από εξαιρετικά μεγάλο δίκτυο ανθρώπινων μυαλών. Η μελλοντική μόδα προβλέπει ότι η μεγαλύτερη ανάπτυξη λογισμικού θα γίνεται στην αποκαλούμενη "Υπερλεωφόρο Πληροφοριών" (Information Super-Highway) που διατρέχει όλο τον κόσμο. Στα χρόνια που έρχονται, η ανάπτυξη του internet θα είναι εκρηκτική η οποία με τη σειρά της θα πυροδοτήσει τη γρήγορη υιοθέτηση της PostgreSQL από τους πελάτες.

Εφαρμόζοντας τις αρχές της φυσικής (κβαντομηχανική, κλασσική, θερμοδυναμική), των μαθηματικών και της στατιστικής στην ποιότητα του λογισμικού, παίρνετε την υψηλότερη ποιότητα λογισμικού μόνο σε ένα 'Ανοικτό Σύστημα Πηγαίου Κώδικα' όπως η PostgreSQL ανοίγοντας τον πηγαίο κώδικα σε ένα πελώριο αριθμό ανθρώπινων μυαλών διασυνδεδεμένων μέσω της Υπερλεωφόρου Πληροφοριών. Όσο μεγαλύτερος ο αριθμός των ανθρώπων που εργάζονται πάνω ς' αυτό, τόσο καλύτερη θα είναι η ποιότητα του λογισμικού. Το Ανοικτό Μοντέλο Πηγαίου Κώδικα θα εμποδίσει επίσης **την ανακάλυψη του τροχού από την αρχή** , θα μειώσει **τη διπλή δουλειά** και θα είναι πολύ οικονομικό, θα εξοικονομεί χρόνο στη διανομή και ακολουθεί τους νόμους της μοντέρνας οικονομίας για βελτιστοποίηση των εθνικών και γενικών πόρων. Όταν μια εργασία λογισμικού έχει γίνει από άλλους, τότε **ΔΕΝ** χρειάζεται να την κάνετε εσείς πάλι. Δε θα σπαταλήσετε τον πολύτιμο χρόνο σας σε κάτι που έχει ήδη γίνει καλά. Καθώς θα μπαίνουμε στον 21ο αιώνα, θα υπάρχει αλλαγή στον τρόπο που παίρνουμε το λογισμικό για χρήση. Οι πελάτες θα προτιμούν τα ανοικτά συστήματα λογισμικού όπως η PostgreSQL, το Linux κλπ.

Αν αγοράσετε εκτελέσιμα προγράμματα, δε θα πάρετε και τον κώδικα και τα δικαιώματά του. Ο πηγαίος κώδικας είναι ανεκτίμητο στοιχείο ενώ τα εκτελέσιμα δεν έχουν καμιά αξία. Το να αγοράσετε λογισμικό *μπορεί* να ανήκει στο παρελθόν. Για να αγοράσετε καλό υλικό (hardware), αξίζει να ξοδέψετε χρήματα και να πάρετε το λογισμικό από το internet. Σημαντικό σημείο είναι το υλικό του υπολογιστή το οποίο **κάνει τον κύριο όγκο της δουλειάς**. Το υλικό είναι το άλογο που κάνει τη δουλειά ενώ το λογισμικό απλά το οδηγεί. Το υλικό των υπολογιστών είναι πολύ πολύπλοκο και μόνο 6 από τις 180 χώρες στον κόσμο μέχρι τώρα έχουν επιδείξει την ικανότητα σχεδιασμού και κατασκευής πλακετών/υλικού υπολογιστών. Ο σχεδιασμός και η κατασκευή chips υπολογιστών είναι προηγμένη τεχνολογία. Είναι μια πολύ σύνθετη διαδικασία, δαπανηρή, απαιτεί μεγάλες επενδύσεις σε εργοστάσια και μηχανές παραγωγής που δουλεύουν με τεχνολογία 0.18 micron. Σε ένα μικρό chip σιλικόνης εκατομμύρια τρανζίστορς/κυκλώματα είναι στενά πακεταρισμένα. Εταιρείες όπως οι AMD, Intel, Cyrix, Hitachi, IBM και άλλες ξόδεψαν σημαντικό αριθμό από ανθρωποχρόνια για να γίνουν κύριοι υψηλών τεχνολογιών όπως η μικροηλεκτρονική (Micro-electronics) και η νανοηλεκτρονική (Nano-electronics). Micro σημαίνει (ένα-εκατομμυριοστό του μέτρου 10ˆ-6), Nano σημαίνει (ένα-δισεκατομμυριοστό του μέτρου 10ˆ-9). Η τρέχουσα τεχνολογία χρησιμοποιεί μικροηλεκτρονική γύρω στα 0.35 micron χρησιμοποιώντας αγωγούς αλουμινίου, και μεγέθη των 0.25 micron χρησιμοποιώντας αγωγούς ηλεκτρονίων από χαλκό. Στο κοντινό μας μέλλον θα χρησιμοποιηθεί τεχνολογία των 0.10 micron με χαλκό και νανοηλεκτρονική για να φτιάξουν τσιπάκια υπολογιστή. Οι αγωγοί αλουμινίου θα αντικατασταθούν από χαλκό στα chips των υπολογιστών, καθώς ο χαλκός είναι καλύτερος αγωγός των ηλεκτρονίων. Στη φωτολιθογραφία, θα χρησιμοποιηθούν τεχνικές υπεριώδους ακτινοβολίας, ακτίνες X ή δέσμης ηλεκτρονίων για χάραξη κυκλωμάτων με μελλοντικά μεγέθη μικρότερα των 0.15 micron.

Σε 20 χρόνια από τώρα, τα τσιπ σιλικόνης θα αντικατασταθούν από μοριακούς υπολογιστές και βιοτσιπς που θα είναι δισεκατομμύρια φορές γρηγορότερα από τα τσιπ σιλικόνης. Τα μόρια είναι ένα σύνολο ατόμων. Και τα άτομα είναι τα μικρά σωματίδια που αποτελούν ό,τι βλέπετε σε αυτό τον κόσμο. Οι μοριακοί υπολογιστές θα χρησιμοποιούν τα μόρια της ύλης σαν υπεργρήγορους διακόπτες on/off. Όταν ο διακόπτης είναι ON σημαίνει 1, και όταν είναι OFF σημαίνει 0. Όλα τα προγράμματα σε αυτόν τον κόσμο βασίζονται στο δυαδικό σύστημα.

Ο παρακάτω πίνακας δείχνει τη πρόοδο και τις μελλοντικές τάσεις προόδου των chips των υπολογιστών.

```
                Advancement of chip capabilities in future
               *******************************************
+--------------------------+---------+---------+---------+---------+--------+---------+
| Item/Year                | 1997    | 1999    | 2001    | 2003    | 2012   | 2020    |
+--------------------------+---------+---------+---------+---------+--------+---------+
| Feature size(micron)     | 0.25    | 0.18    | 0.15    | 0.13    | 0.05   |< 0.00001|
+--------------------------+---------+---------+---------+---------+--------+---------+
| Wafer size(mm)           | 200     | 300     | 300     | 300     | 450    | Mol/Bio |
+--------------------------+---------+---------+---------+---------+--------+---------+
| Min Operating Voltage    | 1.8-2.5 | 1.5-1.8 | 1.2-1.5 | 1.2-1.5 | 0.5-0.6| < 0.001 |
+--------------------------+---------+---------+---------+---------+--------+---------+
| Max power dissipation    | 70      | 90      | 110     | 130     | 175    | 600     |
+--------------------------+---------+---------+---------+---------+--------+---------+
| On-chip frequencey (MHz) | 750     | 1,250   | 1,500   | 2,100   | 10,000 | > 50,000|
+--------------------------+---------+---------+---------+---------+--------+---------+
| DRAM capacity            | 256 MB  | 1 GB    | 2 GB    | 4 GB    | 256 GB | > 1000GB|
+--------------------------+---------+---------+---------+---------+--------+---------+
```

Όπως βλέπετε, το υλικό είναι υψηλής τεχνολογίας και σημαντικό ενώ το λογισμικό είναι λιγότερο δύσκολη τεχνολογία. Αρα, η κατασκευή υλικού/υλικών-αγαθών είναι ζωτικής σημασίας για την εθνική οικονομία! Εταιρείες όπως οι Compaq, Dell, Sun Microsystems, HP, IBM που κατασκευάζουν υπολογιστές είναι οι κύριοι συνεισφέροντες στην οικονομία της Αμερικής σήμερα και στο μέλλον!!

Από την άλλη, κάθε χώρα ανεξαιρέτως στον κόσμο αναπτύσσει/φτιάχνει λογισμικό. Στην πραγματικότητα, οποιοσδήποτε στον κόσμο με ένα μικρό PC χαμηλού-κόστους μπορεί να δημιουργήσει ένα σύστημα μιας βάσης δεδομένων σαν την Oracle . Αλλά θα του έπαιρνε 10 χρόνια (ο διακομιστής της Oracle είναι περίπου 10 ανθρωπο-έτη δουλειά). Ένα ανθρωπο-έτος είναι ένα άτομο που δουλεύει με πλήρη απασχόληση για ένα ολόκληρο έτος. Αν 10 άνθρωποι εργάζονται για 10 έτη τότε έχουν ξοδευτεί 10 ανθρωπο-έτη.

Βάσεις όπως οι Oracle, Informix, Sybase, IBM DB2 (Unix) γράφτηκαν σε γλώσσα "C" και τα εκτελέσιμα δημιουργήθηκαν μεταγλωττίζοντας τον κώδικα και μετά αποστάλθηκαν στους πελάτες. Οι βάσεις Oracle, Sybase, Informix είναι προγράμματα 100 % "C"!!

Μια και έχει γίνει πολλή δουλειά στην PostgreSQL τα τελευταία 12 χρόνια, δεν έχει νόημα ξαναδημιουργήσουμε από την αρχή μια άλλη βάση που να ικανοποιεί το πρότυπο ANSI/ISO SQL. Είναι μεγάλο πλεονέκτημα να χρησιμοποιήσουμε τον υπάρχοντα κώδικα και να προσθέσουμε τα χαρακτηριστικά που λείπουν ή προσθήκες στην PostgreSQL και να αρχίσουμε να τη χρησιμοποιούμε αμέσως.

Η PostgreSQL δεν είναι απλά μια δωρεάν βάση αλλά ένα καλής ποιότητας 'Προϊόν Internet'. Προβλέπεται ότι η ζήτηση προϊόντων "Made By Internet" θα αυξηθεί εκθετικά μια και είναι ικανό να διατηρεί υψηλή ποιότητα, χαμηλό κόστος, εξαιρετικά μεγάλη βάση χρηστών και ατόμων που αναπτύσσουν κώδικα. Αυτά τα έθνη που δε χρησιμοποιούν τα προϊόντα 'Made By Internet' θα χάσουν σοβαρά την "Επανάσταση World-wide Internet" και θα μείνουν πολύ πίσω από τις άλλες χώρες. Ο λόγος είναι ότι το ίδιο το "Internet" είναι η **μεγαλύτερη** εταιρεία παραγωγής λογισμικού σε όλο τον κόσμο!

2 Άλλες Μορφές αυτού του κειμένου

Αυτό το κείμενο είναι τυπωμένο σε 10 διαφορετικές μορφές, ονομαστικά - DVI, Postscript, Latex, LyX, GNU-info, HTML, RTF (Rich Text Format), Plain-text, Σελίδες βοηθείας Unix (Unix man pages) και SGML.

- Μπορείτε να πάρετε αυτό το κείμενο HOWTO σαν ένα αρχείο tar σε μορφή HTML, DVI, Postscript ή SGML από το - `<ftp://sunsite.unc.edu/pub/Linux/docs/HOWTO/other-formats/>`
- Η απλή μορφή κειμένου βρίσκεται στο: `<ftp://sunsite.unc.edu/pub/Linux/docs/HOWTO>`
- Για σχόλια, παρατηρήσεις, διορθώσεις, ενημερώσεις περί της Ελληνικής μετάφρασης αυτού του HOWTO, στείλτε email στην Βούλα Σανιδά : `voulariba@hellug.gr` ή στον Ευριπίδη Παπακώστα : `evris@hellug.gr` και εναλλακτικά : `ep@rocketmail.com`
- Μεταφράσεις σε άλλες γλώσσες όπως Γαλλικά, Γερμανικά, Ισπανικά, Κινέζικα, Γιαπωνέζικα βρίσκονται στο `<ftp://sunsite.unc.edu/pub/Linux/docs/HOWTO>` Οποιαδήποτε βοήθεια για μετάφραση σε άλλες γλώσσες είναι ευπρόσδεκτη.

Το κείμενο γράφτηκε χρησιμοποιώντας τα "εργαλεία SGML" τα οποία θα τα βρείτε στο - `<http://www.xs4all.nl/~cg/sgmltools/>` Αφού μεταγλωττίσετε τον κώδικα θα έχετε εντολές όπως

- sgml2html databasehowto.sgml (για να δημιουργήσετε αρχεία html)
- sgml2rtf databasehowto.sgml (για να δημιουργήσετε αρχεία RTF)
- sgml2latex databasehowto.sgml (για να δημιουργήσετε αρχεία latex)

Το κείμενο βρίσκεται στο -

- `<http://sunsite.unc.edu/LDP/HOWTO/PostgreSQL-HOWTO.html>`

Μπορείτε επίσης να βρείτε το κείμενο στα παρακάτω mirrors sites -

- `<http://www.caldera.com/LDP/HOWTO/PostgreSQL-HOWTO.html>`
- `<http://www.WGS.com/LDP/HOWTO/PostgreSQL-HOWTO.html>`
- `<http://www.cc.gatech.edu/linux/LDP/HOWTO/PostgreSQL-HOWTO.html>`
- `<http://www.redhat.com/linux-info/ldp/HOWTO/PostgreSQL-HOWTO.html>`
- Άλλα mirror sites κοντά σας (network-address-wise) μπορείτε να βρείτε στο `<http://sunsite.unc.edu/LDP/hmirrors.html>` διαλέξτε ένα site και πηγαίνετε στον κατάλογο /LDP/HOWTO/PostgreSQL-HOWTO.html

Για να δείτε το κείμενο σε μορφή dvi, χρησιμοποιήστε το πρόγραμμα xdvi. Το πρόγραμμα xdvi βρίσκεται στο πακέτο tetex-xdvi*.rpm στο Redhat Linux το οποίο βρίσκεται στα κουμπιά ControlPanel | Applications | Publishing | TeX menu. Για να διαβάσετε κείμενο dvi δώστε την εντολή -

xdvi -geometry 80x90 howto.dvi

και αλλάξτε το μέγεθος του παραθύρου με το ποντίκι. Δείτε τη σελίδα βοήθειας για το xdvi.

Για να μετακινηθείτε χρησιμοποιήστε τα βελάκια, τα πλήκτρα Page Up, Page Down, μπορείτε να χρησιμοποιήσετε επίσης τα γράμματα 'f', 'd', 'u', 'c', 'l', 'r', 'p', 'n' για να μετακινηθείτε προς τα πάνω, κάτω, στο κέντρο, στην επόμενη σελίδα, στην προηγούμενη σελίδα κλπ.

Για να απενεργοποιήσετε το expert μενού πατήστε 'x'. Μπορείτε να διαβάσετε αρχεία postscript χρησιμοποιώντας το 'gv' (ghostview) ή 'ghostscript'. Το πρόγραμμα ghostscript βρίσκεται στο πακέτο ghostscript*.rpm και το πρόγραμμα gv στο πακέτο gv*.rpm στο Redhat Linux το οποίο βρίσκεται στα κουμπιά των μενού ControlPanel | Applications | Graphics. Το πρόγραμμα gv είναι πολύ πιο φιλικό στο χρήστη από το ghostscript. Το ghostscript και το gv είναι επίσης διαθέσιμα και σε άλλες πλατφόρμες όπως σε OS/2, Windows 95 και NT, βλέπετε αυτό το κείμενο ακόμη και σε αυτές τις πλατφόρμες.

```
\textgreek{Gia} \textgreek{na} \textgreek{diab'asete} \textgreek{ke'imeno} postscript \textgreek{d
        gv howto.ps
```

```
\textgreek{Gia} \textgreek{na} \textgreek{qrhsimopoi'hsete} \textgreek{to} ghostscript \textgreek{
        ghostscript howto.ps
```

ΠΡΟΣΟΧΗ: Το κείμενο αυτό είναι μεγάλο, ο συνολικός αριθμός σελίδων (σε postscript) αν τυπωθεί θα είναι περίπου 113 σελίδες στα Αγγλικά (στα ελληνικά θα πρέπει να είναι περισσότερες).

Μπορείτε να διαβάσετε το κείμενο σε μορφή HTML χρησιμοποιώντας Netscape Navigator, Microsoft Internet explorer, Redhat Baron Web browser ή οποιοδήποτε από τους άλλους 10 web browsers.

Μπορείτε να διαβάσετε την έξοδο του σε latex, LyX χρησιμοποιώντας το LyX, ένα X-Windows γραφικό πρόγραμμα διασύνδεσης με latex.

3 Οι νόμοι της Φυσικής εφαρμόζονται στο λογισμικό!

Σε αυτό το κεφάλαιο, θα σας δείξουμε πώς η επιστήμη παίζει σημαντικό ρόλο στη δημιουργία διαφόρων αντικειμένων όπως το λογισμικό, το σύμπαν, τα άτομα, την ενέργεια ακόμα και εσάς τους ίδιους! Αυτό το κεφάλαιο θα σας δείξει επίσης γιατί η γνώση της επιστήμης είναι πολύ σημαντική ΠΡΙΝ να ξεκινήσετε να χρησιμοποιείτε τα προϊόντα της. Τα αντικείμενα περιλαμβάνουν ο,τιδήποτε - για παράδειγμα η PostgreSQL, ο χρόνος, η μάζα, η ενέργεια, οι πλανήτες, ο ήλιος, το φεγγάρι, τα αστέρια, οι γαλαξίες, οι άνθρωποι κλπ... είναι αντικείμενα δημιουργήματα της επιστήμης. Αυτό το κεφάλαιο περιγράφει επίσης πώς οι νόμοι της επιστήμης και η στατιστική ευνοούν τα συστήματα open-source code όπως η PostgreSQL και το Linux. Καθώς η ταχύτητα στο διαδίκτυο αυξάνεται καθημερινά, και το διαδίκτυο γίνεται ολοένα και πιο σταθερό, τα συστήματα open-source code γνωρίζουν γρήγορη ανάπτυξη. Και αν οι στατιστικοί κανόνες και οι νόμοι της φυσικής είναι σωστοί, τα συστήματα closed source-code θα αφανιστούν τελικά από τον πλανήτη.

Οι παράγραφοι που ακολουθούν παρακάτω θα σας δείξουν - "πόσο απέραντη είναι η επιστήμη, πόσο σημαντική είναι για τον άνθρωπο και πώς επιδρά σε έργα λογισμικού όπως η PostgreSQL, το Linux".

Το να αναπτύξετε ένα project όπως η PostgreSQL, απαιτεί ενέργεια και χρόνο, άρα η PostgreSQL είναι προϊόν ενέργειας και χρόνου. Μια και η ενέργεια και ο χρόνος μπορούν να εξηγηθούν μόνο επιστημονικά, υπάρχει μια αμφίδρομη σχέση μεταξύ της φυσικής και των έργων λογισμικού όπως η PostgreSQL, το Linux.

Οι επιστημονικοί νόμοι (Φυσική) εφαρμόζονται παντού, συνέχεια, σε ό,τι κάνετε ακόμη και στα έργα λογισμικού όπως συστήματα βάσεων δεδομένων. Η Φυσική εφαρμόζεται ακόμη και όταν μιλάτε (ηχητικά κύματα), περπατάτε (τριβή μεταξύ εδάφους και πέλματος), διαβάζετε ένα βιβλίο ή γράφετε λογισμικό. Πολλοί κλάδοι της επιστήμης όπως η φυσική, η χημεία κλπ. συγχωνεύονται σε μια ευρεία περιοχή, τα Μαθηματικά (επίσης γνωστά και σαν τη Βασίλισσα όλων των Επιστημών). Οτιδήποτε σε αυτό τον κόσμο έχει βαθιές ρίζες στα μαθηματικά, συμπεριλαμβανομένης της PostgreSQL. Η PostgreSQL χρησιμοποιεί τη 'Μοντέρνα Άλγεβρα' ,ένα πολύ μικρό παρακλάδι των μαθηματικών . Η 'Μοντέρνα Άλγεβρα' ασχολείται με τη 'Θεωρία Συνόλων', τη 'Σχεσιακή Άλγεβρα', επιστήμη των Groups, Rings, Collections, Sets, Unions, Intersections, Exclusions, Domains, Lists, κλπ...

Λογισμικό όπως η PostgreSQL υπάρχει σήμερα λόγω της ενέργειας και του χρόνου. Η ενέργεια είναι αυτή που δημιούργησε τον κόσμο, τον ανθρώπινο εγκέφαλο και πολλά άλλα πράγματα! Η μάζα και η ενέργεια είναι η **ΙΔΙΑ** ΟΝΤΟΤΗΤΑ! Το γεγονός ότι η μάζα και η ενέργεια είναι το ίδιο πράγμα ήταν άγνωστο στους ανθρώπους 100 χρόνια πριν!

Οι κυψέλες στον ανθρώπινο εγκέφαλο καταναλώνουν ενέργεια καθώς επεξεργάζονται (παράγουν λογισμικό) μετατρέποντας τη χημική ενέργεια σε θερμική ενέργεια. Ακόμη και τώρα που διαβάζετε αυτή την παράγραφο, οι κυψέλες στον εγκέφαλό σας καίνε βενζίνη και χρησιμοποιούν την ενέργεια. Γι' αυτό ΣΤΑΜΑΤΗΣΤΕ ΝΑ ΔΙΑΒΑΖΕΤΕ ΤΩΡΑ! Η ενέργεια των νευρώνων (κυψέλες στον εγκέφαλο) μπορεί να μετρηθεί στα εργαστήρια. Για παράδειγμα, υπάρχουν πολλά όργανα όπως οι "Ανιχνευτές Ψεμάτων" και άλλα ιατρικά όργανα που μπορούν να μετρήσουν την ενέργεια που καταναλώνει ο εγκέφαλος.

Όλα αυτά υπονοούν ότι ο ανθρώπινος εγκέφαλος είναι μια θερμοδυναμική πηγή ενέργειας. Επειδή ο άνθρωπος είναι μια θερμοδυναμική πηγή ενέργειας, οι νόμοι της θερμοδυναμικής εφαρμόζονται στον εγκέφαλο και έτσι η θερμοδυναμική έχει έμμεσα αποτελέσματα σε λογισμικό όπως η PostgreSQL.

Όπως σε κάθε επιστήμη, είναι αδύνατο να χτίσετε οποιοδήποτε σύστημα ή θεωρία (συμπεριλαμβανομένου ενός συστήματος βάσεων δεδομένων) που να είναι 100 % τέλειο και χωρίς σφάλματα. Είναι σα να κυνηγάτε έναν ανεμοστρόβιλο , δε θα καταφέρουμε **ΠΟΤΕ** να φτιάξουμε το τέλειο σύστημα ή θεωρία. Λεπτομερείς μαθηματικές εξισώσεις/συζητήσεις για την απόδειξη της ύπαρξης ενός 'τέλειου συστήματος' είναι αδύνατο να γίνουν εδώ καθώς ξεπερνά το σκοπό αυτού του κειμένου.

Και μια τέτοια μαθηματική συζήτηση ασχολείται με τον άπειρο αριθμό διαστάσεων που υπάρχουν στη φύση. Δυστυχώς, οι άνθρωποι μπορούν να δουν ή να νιώσουν 4 διαστάσεις αλλά τα μαθηματικά μπορούν να εξερευνήσουν εύκολα και άλλες διαστάσεις.

Άλλες διαστάσεις είναι **'άπειρα μικρότερες'** από τα άτομα και τα ίδια τα άτομα είναι με τη σειρά τους πολύ μικρά και το ανθρώπινο μάτι δε μπορεί να τα ξεχωρίσει! Τα Μαθηματικά είναι πολύ ισχυρά μια και μπορούν να αναλύσουν και να εξηγήσουν τη γέννηση/ θάνατο όλης της υφηλίου. Ο κόσμος μας έχει σχεδόν **μηδενικό μέγεθος** αν τον κοιτάξετε από άλλο σύμπαν και αντίστροφα. Αυτό σημαίνει ότι ο κόσμος μας δεν είναι ορατός (δεν υπάρχει) για τα άτομα που βρίσκονται σε άλλο σύμπαν. Και αντίστροφα! Θεωρητικά, μπορείτε να βγείτε από τον κόσμο μας και να ταξιδέψετε μεγάλες αποστάσεις (δισεκατομμύρια έτη φωτός) σε ελάχιστο χρόνο και να ξαναμπείτε στον κόσμο μας σε ένα διαφορετικό σημείο του χωροχρόνου! Η απόσταση μεταξύ του κόσμου μας και των άλλων κόσμων είναι πραγματικά μηδενικός!

Ακόμη και αν υπάρχει άπειρος αριθμός διαστάσεων, μπορούν όλες να προκύψουν/ δημιουργηθούν από ένα μικρό αριθμό ΚΥΡΙΩΝ διαστάσεων. Άπειρος αριθμός διαστάσεων μπορούν να συνδυαστούν και να καταλήξουν σε κύριες διαστάσεις. Οι κύριες διαστάσεις απλά απορροφούν τις άλλες διαστάσεις χωρίς αυτές να καταστρέφονται. Οι μαθηματικοί χρησιμοποιούν αυτές τις κύριες διαστάσεις για να κατανοήσουν τη γέννηση και το θάνατο του σύμπαντος. Ο κόσμος στον οποίο ζείτε σήμερα δημιουργήθηκε με το **BIG BANG** δισεκατομμύρια χρόνια πριν (σχεδόν 20 δισεκατομμύρια χρόνια πριν) το οποίο δημιουργήθηκε από αλληλεπιδράσεις ατομικών σωματιδίων άλλων διαστάσεων. Λίγο πριν το big bang υπήρχε ένα πολύ μικρό σημείο όπου το μήκος, το πλάτος και το ύψος ήταν ΜΗΔΕΝ (δηλαδή, ο κόσμος μας δεν υπήρχε εκεί!!) ενώ υπήρχε το υπόλοιπο σύμπαν και οι κύριες διαστάσεις. Εκείνη τη χρονική στιγμή δεν υπήρχαν άτομα, αστέρια, πλανήτες και γαλαξίες! Τα άτομα μέσα στο σώμα σας δεν υπήρχαν εκεί! Έχουν γίνει, λοιπόν, τόσα πολλά πράγματα ΠΡΙΝ να γεννηθείτε!

Το Big bang και επομένως και η γέννηση του κόσμου μας προκλήθηκε από λίγα άτομα κύριων διαστάσεων. **ΚΑΠΟΙΟΣ** (κάτι;) προκάλεσε τη συντριβή ελάχιστων πολύ μικρών ατόμων άλλων διαστάσεων με αποτέλεσμα τη δημιουργία του κόσμου μας, και γεννήθηκαν νέες διαστάσεις χρόνου, μήκους, πλάτους, ύψους ! Βλέπουμε το χέρι κάποιου σε αυτή τη διαδικασία. Αυτή η διαδικασία δεν είναι **και πολύ κατανοητή** από τον άνθρωπο.

Ο άνθρωπος προσπαθεί να δημιουργήσει έναν άλλο κόσμο στο εργαστήριο προσομοιώνοντας το γεγονός big bang (βρίσκεται υπό κατασκευή ένας τεράστιος επιταχυντής στην Ευρώπη, άλλη μια κατασκευή στο Ντάλας, η Αμερική σταμάτησε λόγω περικοπών στον προϋπολογισμό). Υπάρχουν άτομα άλλων διαστάσεων, όπως έχουμε κι εμείς άτομα στον δικό μας κόσμο. Θεωρητικά, μπορείτε να δημιουργήσετε/παράγετε άπειρο αριθμό κόσμων! Αυτή η διαδικασία είναι αντιστρεπτή, δηλαδή ο κόσμος μας μπορεί να συρρικνωθεί σε μερικά άτομα άλλων διαστάσεων! Είναι παρόμοιο με το να γεννηθήκατε από δύο μικροσκοπικές κυψέλες οι οποίες θα συγκρουστούν για να δημιουργήσουν μία μόνο κυψέλη. Αυτή η μόνη μικροσκοπική κυψελίδα διαιρέθηκε και πολλαπλασιάστηκε σε 6 τρισεκατομμύρια κυψελίδες μέχρι να γίνει ένας άνθρωπος ύψους 6 πόδια (δηλαδή εσείς!). Υπάρχουν μερικές ομοιότητες μεταξύ ανθρώπου και σύμπαντος, το σύμπαν γεννιέται και αργότερα πεθαίνει όπως και οι άνθρωποι. Μια και είναι αδύνατη μια ΤΕΛΕΙΑ κατάσταση (όπως και η ΑΠΟΛΥΤΩΣ ΜΗ ΤΕΛΕΙΑ κατάσταση), οι κόσμοι σαν τους δικούς μας γεννιούνται και πεθαίνουν αργότερα με μια κυκλική διαδικασία. ΑΛΛΑ δεν υπάρχει ΤΕΛΕΙΟΣ θάνατος, μόνο μετασχηματισμός επιτρέπεται από την επιστήμη! Ο κόσμος μας μεγαλώνει με ταχύτατο ρυθμό, δεν είναι στατικός, αντίθετα είναι πολύ δυναμικός. Αυτός ο κόσμος θα συνεχίσει να μεγαλώνει ώσπου να παρεμβληθεί κάτι που θα καταστρέψει τον κόσμο μας εφαρμόζοντας δυνάμεις με εξωτερικές διαστάσεις που θα απορροφήσουν και θα εξαφανίσουν τον κόσμο μας! Υπάρχουν δύο πιθανότητες - να απομακρυνθεί η μάζα από τον κόσμο μας μέσω των μαύρων τρυπών (αργή διαδικασία), ή να προστεθεί και άλλη μάζα στον κόσμο μας μέσω των άσπρων τρυπών (γρήγορη διαδικασία μεγάλης σύγκρουσης). Μπορεί να υπάρχουν εκατομμύρια κόσμοι αλλά λίγοι είναι αυτοί που δημιουργήθηκαν από κύριες διαστάσεις. Δεν έχετε φτιαχτεί μόνο από άτομα αυτού του κόσμου αλλά και από άτομα άλλων διαστάσεων! Στις μαύρες τρύπες τα άτομα και τα σωματίδια του κόσμου μας μένουν εγκλωβισμένα και μετασχηματίζονται και μετατρέπονται τελείως σε σωματίδια άλλων διαστάσεων! Στο κέντρο μιας μαύρης τρύπας ο χρόνος, το πλάτος και το ύψος είναι μηδέν! Και η μαύρη τρύπα είναι η πόρτα ΕΞΟΔΟΥ/ΕΙΣΟΔΟΥ πρός και από άλλους κόσμους. Πόρτες ΕΙΣΟΔΟΥ/ΕΞΟΔΟΥ προς άλλους κόσμους μπορούν να ανοίξουν παντού, ακόμη και μέσα στο σώμα σας! Μπορεί να υπάρχει άπειρος αριθμός χρωμάτων, γλωσσών υπολογιστή, σχεδίων τσιπ υπολογιστή και θεωρίες αλλά ΔΕΝ ΜΠΟΡΕΙ να υπάρξει ΕΝΑ ΜΟΝΟ ΤΕΛΕΙΟ χρώμα, γλώσσα υπολογιστή, σχέδιο ή σύστημα! Αυτό που μπορείτε να έχετε είναι μόνο ένα ΣΧΕΔΟΝ ΤΕΛΕΙΟ χρώμα (εύρος κύματος), σύστημα, βάση δεδομένων ή θεωρία! Η φύση είναι όπως ένα Καλειδοσκόπιο.

Συνδυάζοντας την ενέργεια εκατομμυρίων ανθρώπων σε όλο τον κόσμο μέσω internet, είναι δυνατό να επιτευχθεί ένα **ΣΧΕΔΟΝ ΤΕΛΕΙΟ** σύστημα (συμπεριλαμβανομένου ενός συστήματος βάσεων δεδομένων). Η ενέργεια του κάθε ανθρώπου ατομικά είναι ελάχιστη, αλλά διασυνδέοντας ένα μεγάλο αριθμό ανθρώπων, η συνολική ενέργεια θα είναι τεράστια για να δημιουργήσει ένα σχεδόν τέλειο σύστημα. Οι άνθρωποι είναι όντα με περιορισμένη ενέργεια και αποθέματα, και πολύ συχνά σε αυτούς οφείλονται λάθη από απλά μέχρι και πολύ σοβαρά. Επειδή η τάση μας για λάθη ποικίλλει σε κάθε άνθρωπο ανάλογα με τις ιδιαίτερες επιδεξιότητες, τη συγκεκριμένη εμπειρία που έχει ο καθένας και τις συγκυρίες της συγκεκριμένης στιγμής, η ανακάλυψη και η ελάττωση των λαθών λογισμικού (bugs) που δημιουργούνται από κάποιο άνθρωπο διευκολύνεται με τη συγκέντρωση πολλών έξυπνων ανθρώπων-μυαλών για να αντιμετωπίσουν το συγκεκριμένο πρόβλημα. Περισσότερα άτομα συνεπάγεται περισσότερες ανθρωποώρες για κώδικα και την αντίστοιχη λειτουργία του.

Είναι ξεκάθαρο ότι το ίντερνετ μπορεί να διασυνδέσει ένα τεράστιο αριθμό ανθρώπων, όπου όλοι μαζί διαθέτουν πολύ ενέργεια και χρόνο. Με το συνδυασμό τους μπορούν να παραχθούν προϊόντα λογισμικού υψηλότερης ποιότητας σε πολύ λιγότερο χρόνο σε σύγκριση με εμπορικές εταιρείες. Ακόμη και μεγάλες εταιρείες όπως η Microsoft, η IBM δε μπορούν να υπερισχύσουν και να αγνοήσουν τους νόμους της Φυσικής αλλά θα **ΥΠΟΤΑΧΘΟΥΝ** τελικά σε αυτούς!

Σήμερα, υπάρχουν τόσες πολλές βάσεις δεδομένων σε όλο τον κόσμο που στοχεύουν μόνο στις προδιαγραφές ANSI/ISO SQL. **Οι άνθρωποι δεν πρέπει να χαραμίζουν το χρόνο τους για να δημιουργήσουν τόσα πολλά ΠΑΡΟΜΟΙΑ/ΟΛΟΙΔΙΑ πακέτα λογισμικού** ενώ υπάρχουν τόσοι άλλοι τομείς της επιστήμης που χρειάζονται προσοχή. Το συμπέρασμα είναι το εξής : εξαιτίας των νόμων της επιστήμης, τα 'ανοιχτά συστήματα πηγαίου κώδικα' όπως η PostgreSQL, το Linux θα είναι πάντα πολύ καλύτερα από τα 'κλειστά συστήματα πηγαίου κώδικα' και αυτό είναι δυνατό να αποδειχτεί και επιστημονικά.

4 Τι είναι η PostgreSQL ;

Η PostgreSQL έκδοση 6.4 είναι μια δωρεάν βάση, με την οποία σας δίνεται όλος ο πηγαίος κώδικας και είναι ένα Σύστημα Αντικειμενοστραφής-Σχεσιακή Βάσης Δεδομένων σχεδόν συμβατή με (πλησιάζει όλο και περισσότερο) με το ANSI SQL1998,92,89 και τρέχει σε διαφορετικές πλατφόρμες hardware και Λειτουργικά συστήματα.

```
\textgreek{Merik'ec} \textgreek{for'ec} \textgreek{bga'inoun} \textgreek{se} \textgreek{kuklofor'i
\textgreek{laj'wn} \textgreek{met'a} \textgreek{thn} \textgreek{'ekdosh} GA PostgreSQL. \textgreek
\textgreek{an'aloga} \textgreek{me} \textgreek{tic} \textgreek{an'agkec} \textgreek{thc} \textgree
\textgreek{b'hmata} \textgreek{gia} \textgreek{na} \textgreek{efarm'osete} \textgreek{ta} patches
\textgreek{Phga'inete} \textgreek{ston} \textgreek{kat'alogo} \textgreek{pou} \textgreek{br'isketa
        cd /usr/src/postgresql6.3.1
        patch -p0 < patchfile
        make clean
        make
\textgreek{De'ite} \textgreek{ep'ishc} \textgreek{th} \textgreek{sel'ida} \textgreek{bo'hjeiac} \t
```

Τα αρχεία για patch βρίσκονται στο

- PostgreSQL patches : <ftp://ftp.postgresql.org/pub/patches>

Ο απόλυτος σκοπός και ο τελικός στόχος της PostgreSQL είναι γίνει 100 % συμβατή με την ANSI/ISO SQL και να γίνει επίσης η νούμερο ENA Ανοικτή Γενική Βάση στον κόσμο. Η PostgreSQL επίσης θα οδηγήσει, διευθύνει, ελέγξει, παρακολουθήσει και υπαγορεύσει το μέλλον της ANSI/ISO SQL. Αυτό γιατί, η υλοποίηση και οι ιδέες γίνονται πρώτα στην PostgreSQL και έπειτα ενσωματώνονται στην ANSI/ISO SQL.

Ο Informix Universal server (εκδόθηκε το 1997) βασίζεται σε προηγούμενη έκδοση της PostgreSQL επειδή η Informix αγόρασε την Illustra Inc. και ενοποιήθηκε με την Informix. Η βάση Illustra ήταν πλήρως βασισμένη στην Postgres (προηγούμενη έκδοση της PostgreSQL).

Η PostgreSQL είναι μια επέκταση του συστήματος διαχείρισης της βάσης δεδομένων POSTGRES, ένα επόμενης γενιάς πρότυπο έρευνας DBMS. Ενώ η PostgreSQL διατηρεί το δυναμικό μοντέλο δεδομένων και τους πλούσιους τύπους δεδομένων της POSTGRES, αντικαθιστά την γλώσσα ερωτήσεων PostQuel με ένα εκτεταμένο υποσύνολο της SQL.

Η ανάπτυξη της PostgreSQL γίνεται από μια ομάδα προγραμματιστών στο Internet που είναι όλοι γραμμένοι στη λίστα αλληλογραφίας ανάπτυξης της PostgreSQL. Συντονιστής της ομάδας είναι ο Marc G. Fournier

- scrappy@postgreSQL.org

Αυτή η ομάδα είναι υπεύθυνη για όλη την τρέχουσα και μελλοντική ανάπτυξη της PostgreSQL. Φυσικά, ο ίδιος ο πελάτης βάσεων δεδομένων είναι αυτός που αναπτύσσει την PostgreSQL! Το φορτίο της ανάπτυξης διανέμεται σε ένα πολύ μεγάλο αριθμό τελικών χρηστών βάσεων στο internet.

Οι συγγραφείς της PostgreSQL 1.01 είναι οι Andrew Yu και Jolly Chen. Πολλοί άλλοι συνεισέφεραν σε αυτή τη μεταφορά, το τεστάρισμα, την αποσφαλμάτωση και την επέκταση του κώδικα. Ο αρχικός κώδικας της Postgres, απ' όπου προέκυψε η PostgreSQL, ήταν η προσπάθεια πολλών αποφοίτων σπουδαστών, μεταπτυχιακών, και προσωπικού προγραμματιστών που δούλεψαν κάτω από την επίβλεψη του Καθηγητή Michael Stonebraker στο Πανεπιστήμιο της Καλιφόρνια, στο Berkeley.

Το αρχικό όνομα του λογισμικού στο Berkeley ήταν Postgres. Όταν προστέθηκε η λειτουργικότητα για SQL το 1995, το όνομα άλλαξε σε Postgres95. Το όνομα άλλαξε πάλι στο τέλος του 1996 σε PostgreSQL.

Εκατομμύρια PostgreSQL έχουν εγκατασταθεί ως διακομιστές βάσεων δεδομένων, Web database serversκαι Application data servers. Είναι πολύ προχωρημένη, και είναι αντικειμενοστραφής σχεσιακή βάση δεδομένων (ORDBMS).

Η PostgreSQL μπορεί να αποθηκεύσει περισσότερους τύπους δεδομένων από τους παραδοσιακούς τύπους δεδομένων όπως integer, characters, κλπ. - μπορείτε να δημιουργήσετε τους δικούς σας ορισμένους τύπους, συναρτήσεις, κληρονομικότητα κλπ. Η PostgreSQL τρέχει σε Solaris, SunOS, HPUX, AIX, Linux, Irix, Digital Unix, BSDi,NetBSD, FreeBSD, SCO unix, NEXTSTEP, Unixware και σε οποιοδήποτε τύπο Unix. Η μεταφορά της σε Windows 95/NT είναι σε εξέλιξη.

- Τίτλος: PostgreSQL SQL RDBMS Database (Σύστημα Διαχείρισης Αντικειμενοστραφούς Σχεσιακής Βάσης Δεδομένων)
- Τρέχουσα έκδοση: 6.4
- Χρονολογία: Η PostgreSQL είναι 12 χρονών. Αναπτύσσεται από το 1985
- Συγγραφείς: Αναπτύσσεται από εκατομμύρια/πανεπιστήμια/εταιρείες στο internet για τα τελευταία 12 χρόνια

Η PostgreSQL και τα σχετικά κομμάτια του κειμένου υπόκεινται στην παρακάτω ΠΝΕΥΜΑΤΙΚΗ ΙΔΙΟΚΤΗΣΙΑ.

```
PostgreSQL \textgreek{S'usthma} \textgreek{Diaqe'irishc} \textgreek{B'ashc} \textgreek{Dedom'enwn}

\textgreek{Pneumatik'h} \textgreek{Idiokths'ia} (c) \textgreek{tou} \textgreek{panepisthm'iou} \textgreek{

\textgreek{'Adeia} \textgreek{gia} \textgreek{qr'hsh}, \textgreek{antigraf'h}, \textgreek{metabol'h}, \tex
\textgreek{tekmhr'iws'hc} \textgreek{tou} \textgreek{gia} \textgreek{opoiod'hpote} \textgreek{skop'o}, \te
\textgreek{par'eqetai}, \textgreek{me} \textgreek{thn} \textgreek{pro"up'ojesh} \textgreek{'oti} \textgree
\textgreek{aut'h} \textgreek{kai} \textgreek{oi} \textgreek{parak'atw} \textgreek{d'uo} \textgreek{par'agr

\textgreek{ME} \textgreek{KANENAN} \textgreek{TROPO}, \textgreek{DEN} \textgreek{JA} \textgreek{JEWRHJEI}
\textgreek{GIA} \textgreek{AMESH}, \textgreek{EMMESH}, \textgreek{EIDIKH}, \textgreek{TUQAIA}, \textgreek{
\textgreek{APWLEIAS} \textgreek{KERDWN} \textgreek{POU} \textgreek{PROEKUYE} \textgreek{APO} \textgreek{TH
\textgreek{TEKMHRIWSHS} \textgreek{TOU}, \textgreek{AKOMA} \textgreek{KAI} \textgreek{AN} \textgreek{TO} \
\textgreek{THN} \textgreek{PIJANOTHTA} \textgreek{MIAS} \textgreek{TETOIAS} \textgreek{ZHMIAS}.

\textgreek{TO} \textgreek{PANEPISTHMIO} \textgreek{THS} \textgreek{KALIFORNIA} \textgreek{EIDIKA} \textgre
\textgreek{ALLA} \textgreek{OQI} \textgreek{PERIORISMENHS} \textgreek{MONO} \textgreek{SE} \textgreek{AUTH
\textgreek{GIA} \textgreek{KAPOIO} \textgreek{SUGKEKRIMENO} \textgreek{SKOPO}. \textgreek{TO} \textgreek{L
\textgreek{KAI} \textgreek{TO} \textgreek{PANEPISTHMIO} \textgreek{THS} \textgreek{KALIFORNIAS} \textgreek
\textgreek{EMPLOUTISMWN}, \textgreek{H} \textgreek{TROPOPOIHSEWN}.
```

5 Πού μπορώ να τη βρω ;

Μπορείτε να αγοράσετε το CDROM Redhat Linux, Debian Linux ή το Slackware Linux που περιέχουν ήδη την Postgresql σε μορφή πακέτου (και ο πηγαίος κώδικας και τα εκτελέσιμα) από τα :

- Linux System Labs Web site: `<http://www.lsl.com/>` 7 (U.S. δολάρια)
- Cheap Bytes Inc Web site: `<http://www.cheapbytes.com/>` 7 (U.S. δολάρια)
- Debian Κεντρικό Web site : `<http://www.debian.org/vendors.html>`

Ο οργανισμός PostgreSQL πουλά επίσης ένα 'PostgreSQL CDROM' που περιέχει όλον τον πηγαίο κώδικα και εκτελέσιμα για πολλά λειτουργικά συστήματα Unix και πλήρη τεκμηρίωση.

- Το CDROM της PostgreSQL θα το βρείτε από το κεντρικό Web site της στο : `<http://www.postgresql.org>` για 29 (Αμερικάνικα δολάρια)

Διανομή της PostgreSQL μόνο με εκτελέσιμα:

- Μπορείτε να τρέξετε την PostgreSQL χωρίς να μεταγλωττίσετε τον κώδικα. Εγκαταστήστε τα εκτελέσιμα για Intel-Linux από το αρχείο `<ftp://www.redhat.com/pub/contrib/i386/>` postgresql-6.4.i386.rpm. Αυτό είναι ένα redhat 'rpm' πακέτο και περιέχει και τον κώδικα και τα εκτελέσιμα για την PostgreSQL.
- Site με εκτελέσιμα για Solaris, HPUX, AIX, IRIX, Linux : `<ftp://ftp.postgresql.org/pub/bindist>` Αν μεταγλωττίσετε τον κώδικα για οποιαδήποτε πλατφόρμα, παρακαλώ στείλτε το στο site, μπορεί να είναι χρήσιμο για τους υπόλοιπους.
- ftp site : Πάρτε τα εκτελέσιμα για Intel-Linux από το αρχείο `<ftp://ftp.redhat.com/pub/contrib/i386/>` postgresql-6.4.i386.rpm. Αυτό είναι ένα redhat 'rpm' πακέτο και περιέχει και τον κώδικα και τα εκτελέσιμα για την PostgreSQL.

WWW Web sites:

- Κεντρικό Web site: `<http://www.postgresql.org/>`
- Δεύτερο Web site: `<http://logical.thought.net/postgres95/>`
- `<http://www.itm.tu-clausthal.de/mirrors/postgres95/>`
- `<http://s2k-ftp.cs.berkeley.edu:8000/postgres95/>`
- `<http://xenium.pdi.net/PostgreSQL/>`
- `<http://s2k-ftp.cs.berkeley.edu:8000/postgres95/>`

Τα ftp sites παρουσιάζονται παρακάτω :-

- Primary FTP: `<ftp://ftp.postgresql.org/pub>`
- Secondary FTP: `<ftp://ftp.chicks.net/pub/postgresql>`
- `<ftp://ftp.emsi.priv.at/pub/postgres/>`
- `<ftp://ftp.itm.tu-clausthal.de/pub/mirrors/postgres95>`
- `<ftp://rocker.sch.bme.hu/pub/mirrors/postgreSQL>`
- `<ftp://ftp.jaist.ac.jp/pub/dbms/postgres95>`
- `<ftp://ftp.luga.or.at/pub/postgres95>`
- `<ftp://postgres95.vnet.net:/pub/postgres95>`
- `<ftp://ftpza.co.za/mirrors/postgres>`
- `<ftp://sunsite.auc.dk/pub/databases/postgresql>`
- `<ftp://ftp.task.gda.pl/pub/software/postgresql>`
- `<ftp://xenium.pdi.net/pub/PostgreSQL>`

Ο πηγαίος κώδικας της PostgreSQL είναι επίσης διαθέσιμος σε όλα τα mirror sites του sunsite unc (σύνολο περίπου 1000 sites σε όλο τον κόσμο). Βρίσκεται επίσης στη διανομή του Red Hat Linux στο αρχείο /pub/contrib/i386/postgresql.rpm .

- Για τη λίστα των mirror sites πηγαίνετε στο `<ftp://sunsite.unc.edu>`

6 Η PostgreSQL υποστηρίζει εξαιρετικά μεγάλες βάσεις, μεγαλύτερες από 200 Gig

Η απόδοση των 32-bit μηχανών cpu θα πέσει γρήγορα όταν το μέγεθος της βάσης ξεπεράσει τα 5 gig. Μπορείτε να τρέξετε βάση των 30 gig σε μια 32-bit cpu αλλά η απόδοση δε θα είναι και πολύ εντυπωσιακή. Υπάρχουν περιορισμοί που επιβάλλονται από το μέγεθος της μνήμης RAM (μέγιστο 2 gig μόνο), περιορισμοί από τα μεγέθη του συστήματος αρχείων και επίσης από το λειτουργικό σύστημα.

Αν χρειάζεται να χρησιμοποιήσετε εξαιρετικά μεγάλες βάσεις, σας συμβουλεύουμε να χρησιμοποιήσετε μηχανές των 64-bit όπως Digital Alpha cpu, Sun Ultra-sparc 64-bit cpu, Silicon graphics 64-bit cpu, Intel Merced IA-64 cpu, μηχανές HPUX 64bit, IBM 64-bit. Μεταγλωττίστε την PostgreSQL σε μια 64-bit cpu και θα μπορεί να υποστηρίζει τεράστιες βάσεις και μεγάλα queries. Η απόδοση της PostgreSQL για ερωτήσεις σε μεγάλους πίνακες και βάσεις θα είναι πολλές φορές ταχύτερη από την PostgreSQL σε μηχανές με cpu 32-bit. Το πλεονέκτημα των μηχανών 64-bit είναι ότι η δεικτοδότηση του χώρου μνήμης είναι πολύ μεγάλη και το λειτουργικό σύστημα μπορεί να υποστηρίξει συστήματα μεγάλων αρχείων, παρέχει καλύτερη απόδοση με μεγάλες βάσεις, υποστηρίζει πολύ περισσότερη μνήμη (RAM), έχουν περισσότερες δυνατότητες κλπ.

7 Πώς μπορώ να εμπιστευθώ την PostgreSQL ; Το πακέτο Regression για τεστ αναπτύσσει το αίσθημα εμπιστοσύνης στον πελάτη

Το πακέτο Regression για τεστ (src/test/regress) περιλαμβάνεται στη διανομή και αφήστε τον υπολογιστή σας να επαληθεύσει τις στάνταρ λειτουργίες SQL και τις δυνατότητες επέκτασης της PostgreSQL. Το πακέτο για τεστ περιέχει ήδη εκατοντάδες SQL τεστ προγράμματα.

Θα πρέπει να χρησιμοποιήσετε τη δύναμη του υπολογιστή για να ελέγξετε την PostgreSQL, αντί να χρησιμοποιήσετε τη δύναμη του ανθρώπινου εγκεφάλου. Οι υπολογιστές μπορούν να εκτελέσουν τεστ για το λογισμικό regression εκατομμύρια ή ακόμη και δισεκατομμύρια φορές γρηγορότερα από ότι οι άνθρωποι.

Στο προσεχές μέλλον η ταχύτητα των υπολογιστών θα είναι αρκετά τρισεκατομμύρια φορές γρηγορότεροι από τον ανθρώπινο εγκέφαλο!

Έτσι, έχει νόημα να χρησιμοποιήσουμε τη δύναμη του υπολογιστή για να ελέγχουμε λογισμικό.

Μπορείτε να προσθέσετε και άλλα τεστ σε περίπτωση που το χρειάζεστε, και μπορείτε να τα στείλετε στο κεντρικό web site της PostgreSQL αν νομίζετε ότι θα είναι χρήσιμο σε άλλους στο internet. Το πακέτο Regression για τεστ ενισχύει την εμπιστοσύνη του πελάτη στην PostgreSQL και ευκολύνει τη γρήγορη εφαρμογή της PostgreSQL σε συστήματα παραγωγής.

Το πακέτο Regression για τεστ μπορεί να θεωρηθεί σαν ένα **"ΠΟΛΥ ΣΤΑΘΕΡΟ"** τεχνικό κείμενο συμφωνημένο αμοιβαία μεταξύ των προγραμματιστών και των τελικών χρηστών. Οι προγραμματιστές της PostgreSQL χρησιμοποιούν εκτεταμένα το πακέτο regression για τεστ κατά την περίοδο της ανάπτυξης και επίσης πριν την έκδοση του λογισμικού στο κοινό για να διασφαλίσουν την καλή ποιότητα.

Οι δυνατότητες της PostgreSQL αντικατροπτίζονται στο πακέτο για τεστ regression. Σε περίπτωση που μια δυνατότητα, σύνταξη ή χαρακτηριστικό δεν περιλαμβάνεται στο πακέτο δεν σημαίνει πως καλύπτεται αναγκαστικά από την PostgreSQL. Μπορείτε να το ελέγξετε, παρόλα αυτά, εσείς και να ενσωματώσετε το τεστ σας στο πακέτο.

8 Γραφικό πρόγραμμα επικοινωνίας (GUI FrontEnd Tool) με την PostgreSQL

Η διανομή της PostgreSQL περιλαμβάνει μια βιβλιοθήκη σε Tcl/Tk, που ονομάζεται 'pgTcl'. Η Tcl/Tk είναι ένα εργαλείο για γρήγορη ανάπτυξη εφαρμογών και μια πολύ δυνατή scripting γλώσσα. Αναπτύσσετε μια φορά την εφαρμογή σας και την τρέχετε παντού : σε NT, Win 95, Linux, OS/2 και όλα τα συστήματα unix. Tcl είναι τα αρχικά του 'Tool Command Language' και Tk του 'Tool Kit'. Υπάρχει ένα ολοκληρωμένο περιβάλλον ανάπτυξης (IDE - integrated development environment) για την Tcl/Tk που λέγεται SpecTcl. Κοιτάξτε γι αυτό το rpm πακέτο στη διανομή Redhat linux ή στο sunscript site που δίνεται παρακάτω.

Συνιστούμε την Perl για γλώσσα προγραμματισμού μια και είναι η πιο διαδεδομένη γλώσσα, πολύ ισχυρή και διατίθεται για όλα τα μηχανήματα και όλες τις πλατφόρμες λειτουργικών. Σας συμβουλεύουμε να χρησιμοποιήσετε Perl-Tk ή Perl-Qt ή C++ QtEZ ή C++ Lesstiff. Ο συνδυασμός Perl, C και Tk θα ικανοποιήσει τις ανάγκες των περισσοτέρων projects, λαμβάνοντας υπόψη το διαθέσιμο χρόνο, την ταχύτητα των εκτελέσιμων και την ευκολία προγραμματισμού. Σημειώστε ότι τα προγράμματα σε Java τρέχουν πολύ αργά. Ακόμη και τα προγράμματα που είναι γραμμένα σε C++ τρέχουν πιο αργά από τα προγράμματα που γράφονται σε C. Τα προγράμματα σε C εκτελούνται ταχύτερα από τα αντίστοιχα προγράμματα γραμμένα σε οποιαδήποτε άλλη γλώσσα προγραμματισμού.

Η Perl χρησιμοποιείται για τον προγραμματισμό του 90 % των internet scripts στον κόσμο. Έτσι θα έχετε μόνο μια γλώσσα, την "Perl", για όλες σας τις ανάγκες, σε εφαρμογές όσο και στο internet. Δείτε το Perl Database Interface αυτού του κειμένου στο 11 ()

Θα πρέπει να μελετήσετε τα δυνατά σημεία της Perl σε σύγκριση με την Tcl/Tk πριν ξεκινήσετε κάποιο project σε Tcl. Η Tcl/TK δίνεται συνήθως με κάθε linux cdrom. Μπορείτε επίσης να τη βρείτε στα sites -

- Tcl/Tk `<http://www.scriptics.com>`
- Αντικειμενοστραφές extension της Tcl που ονομάζεται INCR στο `<http://www.tcltk.com>`
- Visual TCL site `<http://www.neuron.com>`
- Visual TCL Redhat rpm στο `<ftp://ftp.redhat.com/pub/contrib/i386/visualtcl*.rpm>`
- `<http://sunscript.sun.com/>`
- `<http://sunscript.sun.com/TclTkCore/>`
- `<ftp://ftp.sunlabs.com/pub/tcl/tcl8.0a2.tar.Z>`
- Βιβλία αναφορών: Πολλά τεχνικά βιβλία για TCL/TK διατίθενται στην αγορά.

9 Εργαλεία ενός Ολοκληρωμένου Συστήματος Ανάπτυξης για PostgreSQL (GUI IDE)

Κοιτάξτε τα παρακάτω εργαλεία ανάπτυξης τα οποία μπορείτε να χρησιμοποιήσετε σε συνδυασμό με οδηγούς ODBC/JDBC. Αυτοί είναι παρόμοιοι με Borland C++ Builder, Borland JBuilder.

Το Vibe είναι Java και C++ IDE (Integrated Development Environment) και κέρδισε στην Unix Επισκόπηση για τα ολοκληρωμένα συστήματα ανάπτυξης της χρονιάς.

- Περισσότερες πληροφορίες για το Vibe στο `<http://www.LinuxMall.com/products/00487.html>`
- Το 'QT' είναι μια πλατφόρμα ανάπτυξης εφαρμογών για Windows95/NT και Unix `<http://www.troll.no>` και το ftp site του είναι το `<ftp://ftp.troll.no>`

Μπορείτε επίσης να χρησιμοποιήσετε Borland C++ Builder, Delphi, Borland JBuilder, PowerBuilder σε Windows95 και να συνδεθείτε με την PostgreSQL σε ένα σύστημα unix μέσω οδηγών ODBC/JDBC.

Δείτε επίσης το **PERL** Database Interface σε αυτό το κείμενο στο 11 () H Perl και C/C++ συνίστανται έντονα σαν επιλογές γλώσσας για IDE.

Δωρεάν εργαλεία IDE -

Δείτε το CDROM της διανομής redhat linux.

- Lesstiff Mottif Tool `<ftp://ftp.redhat.com/pub/contrib/i386/lesstiff*.rpm>`
- FreeBuilder `<ftp://ftp.redhat.com/pub/contrib/i386/free*.rpm>`
- SpecTCL `<ftp://ftp.redhat.com/pub/contrib/i386/spec*.rpm>`
- JccWarrior `<ftp://ftp.redhat.com/pub/contrib/i386/jcc*.rpm>`
- Kanchenjunga Java RAD Tool for PostgreSQL `<http://www.man.ac.uk/~whaley/kj/kanch.html>`
- Applixware Tool `<http://www.redhat.com>`
- XWPE X Windows Programming Environment `<http://www.rpi.edu/~payned/xwpe/>` `<ftp://ftp.redhat.com/pub/contrib/i386/xwpe*.rpm>`
- XWB X Windows Work Bench `<ftp://ftp.redhat.com/pub/contrib/i386/xwb*.rpm>`
- NEdit `<ftp://ftp.redhat.com/pub/contrib/i386/nedit*.rpm>`

10 Οδηγοί Διασύνδεσης με την PostgreSQL

10.1 ODBC Drivers για την PostgreSQL

ODBC είναι ακρωνύμιο του 'Open DataBase Connectivity', ένα δημοφιλές πρότυπο για πρόσβαση σε πληροφορίες από διάφορες βάσεις από διαφορετικούς πωλητές. Οι εφαρμογές που έχουν γραφτεί χρησιμοποιώντας ODBC drivers, εγγυούνται ότι θα δουλέψουν με διάφορες βάσεις όπως PostgreSQL, Oracle, Sybase, Informix κλπ.

- Το PostODBC έχει συμπεριληφθεί ήδη στη διανομή. Δείτε το κεντρικό web site `<http://www.postgresql.org>`. Συμπεριλαμβάνεται στο CDROM της PostgreSQL.
- `<http://www.openlinksw.com>` H Open Link Software Corporation πουλάει ODBC για PostgreSQL και άλλες βάσεις. H Open Link δίνει επίσης δωρεάν ODBC (περιορισμένο αριθμό χρηστών) - τσεκάρετέ το.
- Insight ODBC για την PostgreSQL `<http://www.insightdist.com/psqlodbc>` Αυτό είναι το επίσημο site του PostODBC.

Υπάρχει κάποιο έργο που ονομάζεται FreeODBC Pack Package. Δεν υπάρχει ς' αυτό έκδοση για PostgreSQL, ίσως μπορείτε να βοηθήσετε.

- `<http://www.ids.net/~bjepson/freeODBC/>` Αυτή είναι η δωρεάν έκδοση για ODBC.

10.2 UDBC Drivers για την PostgreSQL

Το UDBC είναι μια στατική έκδοση ODBC ανεξάρτητη από διαχειριστές οδηγών (driver managers) και υποστήριξη DLL, χρησιμοποιείται για να προσθέτει υποστήριξη απευθείας διασύνδεσης με βάση μέσα στις εφαρμογές.

- <http://www.openlinksw.com>Η εταιρεία Open Link Software πουλά UDBC για την PostgreSQL και άλλες βάσεις. Η Open Link δίνει επίσης δωρεάν UDBC (περιορισμένο αριθμό χρηστών) - τσεκάρετέ το.

10.3 JDBC Drivers για την PostgreSQL

JDBC είναι ακρωνύμιο του 'Java DataBase Connectivity'. Η Java είναι μια γλώσσα προγραμματισμού ανεξαρτήτου πλατφόρμας που αναπτύχθηκε από την Sun Microsystems. Οι προγραμματιστές Java ενθαρρύνονται να γράφουν εφαρμογές βάσεων δεδομένων χρησιμοποιώντας JDBC για να διευκολύνουν τη μεταφερσιμότητα μεταξύ βάσεων όπως η PostgreSQL, Oracle, informix, κλπ. Αν γράψετε εφαρμογή σε Java μπορείτε να πάρετε τους οδηγούς JDBC για PostgreSQL από τα παρακάτω sites:

Ο οδηγός JDBC περιλαμβάνεται ήδη στη διανομή της PostgreSQL.

- <http://www.demon.co.uk/finder/postgres/index.html> Διασύνδεση Java (της Sun) με PostgreSQL
- <ftp://ftp.ai.mit.edu/people/rst/rst-jdbc.tar.gz>
- <http://www.openlinksw.com> Η εταιρεία Open Link Software πουλά JDBC για PostgreSQL και άλλες βάσεις. Η Open Link δίνει επίσης δωρεάν JDBC (περιορισμένο αριθμό χρηστών) - τσεκάρετέ το.
- JDBC UK site <http://www.retep.org.uk/postgres>
- JDBC FAQ site <http://eagle.eku.edu/tools/jdbc/faq.html>

Τα JDBC home, guide και FAQ βρίσκονται στα -

- JDBC HOME <http://splash.javasoft.com/jdbc>
- JDBC guide <http://www.javasoft.com/products/jdk/1.1/docs/guide/jdbc>
- JDBC FAQ <http://javanese.yoyoweb.com/JDBC/FAQ.txt>

10.4 Java για PostgreSQL

Οι προγραμματιστές Java μπορεί να θεωρήσουν τα παρακάτω sites πολύ χρήσιμα για την PostgreSQL.

- <ftp://ftp.redhat.com/pub/contrib/i386> και δείτε το postgresql-jdbc-*.rpm
- <http://www.blackdown.org>

11 Perl Database Interface (DBI) Driver για PostgreSQL

11.1 Διασύνδεση Perl 5 με PostgreSQL

PERL είναι ακρωνύμιο του 'Practical Extraction και Report Language'. Η Perl είναι διαθέσιμη σχεδόν σε κάθε λειτουργικό και πλατφόρμα hardware. Μπορείτε να χρησιμοποιήστε Perl σε Windows95/NT, όλα τα

Unix (Solaris, HPUX, AIX, Linux, Irix, SCO κλπ.), σε mainframe MVS, desktop OS/2, OS/400, Amdahl UTS και πολλά άλλα. Η Perl τρέχει **ΑΚΟΜΗ** και σε πολλά μη δημοφιλή/γενικά άγνωστα λειτουργικά συστήματα και υλικά!!

Έτσι, μην εκπλαγείτε αν δείτε την perl να τρέχει σε πολλά όχι συχνά χρησιμοποιούμενα λειτουργικά.

Μπορείτε να φανταστείτε την τεράστια βάση χρηστών και προγραμματιστών σε Perl.

Η διασύνδεση Perl με PostgreSQL περιλαμβάνεται στη διανομή της PostgreSQL. Κοιτάξτε στον κατάλογο src/pgsql_perl5.

- Το Email για επικοινωνία σχετικά με την Pgsql_perl5 : `E.Mergl@bawue.de`
- Άλλη πηγή είναι τα - `<ftp://ftp.kciLink.com/pub/PostgresPerl-1.3.tar.gz>`
- Perl Home page: `<http://www.perl.com/perl/index.html>`
- Εκμάθηση της Perl, ψάξτε για τίτλο Tutorial στο : `<http://reference.perl.com/>`
- Το Perl FAQ είναι στο : `<http://www.yahoo.com/Computers_and_Internet/Programming_Languages/Perl/>`
- Γραφικό περιβάλλον για Perl, Perl-Qt : `<ftp://ftp.redhat.com/pub/contrib/i386>` και ψάξτε για το PerlQt-1.06-1.i386.rpm
- Γραφικό περιβάλλον για Perl, Perl-Qt : `<http://www.accessone.com/~jql/perlqt.html>`
- Γραφικό περιβάλλον για Perl, Perl-XForms : `<ftp://ftp.redhat.com/pub/contrib/i386>` και ψάξτε για το αρχείο Xforms4Perl-0.8.4-1.i386.rpm
- Γραφικό περιβάλλον για Perl, Perl-Tk : `<ftp://ftp.redhat.com/pub/contrib/i386>`
- Perl GUIkits : `<http://reference.perl.com/query.cgi?ui>`
- Perl Database Interfaces : `<http://reference.perl.com/query.cgi?database>`
- Μεταφραστής από Perl σε "C" : `<http://www.perl.com/CPAN-local/modules/by-module/B/>` και ψάξτε για το Compiler-a3.tar.gz
- Μεταφραστής από Perl σε κέλυφος Bourne : `<http://www.perl.com/CPAN/authors/id/MERLYN/sh2perl-0.02.tar.gz>`
- από awk σε perl a2p και από sed σε perl s2p περιλαμβάνεται με τη διανομή της PERl.
- Δείτε επίσης τα newsgroups για PERL στο comp.lang.perl.*

11.2 Perl Database Interface DBI

11.2.1 TI EINAI TO DBI ;

Το παρακάτω είναι **απόσπασμα** από την κεντρική σελίδα του Perl DBI.

Το Perl Database Interface (DBI) είναι ένα προγραμματιστικό interface (API) σε Perl για διασύνδεση με βάσεις δεδομένων. Οι προσδιορισμοί του Perl DBI API ορίζουν ένα σύνολο από ρουτίνες, μεταβλητές και συμβάσεις που παρέχουν ένα συνεκτικό interface βάσης δεδομένων ανεξάρτητο από την πραγματική βάση που χρησιμοποιείται. Οι πληροφορίες για αυτό το τμήμα DBI προήλθε από το κείμενο 'DBI FAQ' το οποίο έχει γράψει ο Alligator Descartes το αναπαραγάγαμε με την άδειά του.

- Μπορείτε να επικοινωνήσετε με τον Alligator Descartes Hermetica στο `descarte@hermetica.com`

11.2.2 Οδηγός DBI για την PostgreSQL - DBD-Pg-0.89

Κατεβάστε το DBD-Pg-0.89.tar.gz από τα

- DBD-Pg-0.89 : `<http://www.perl.com/CPAN/modules/by-module/DBD/>`
- Comprehensive Perl Archive Network CPAN `<http://www.perl.com/CPAN>`
- Λίστα οδηγών DBI και σελίδες DBI module `<http://www.hermetica.com/technologia/perl/DBI>`
- Πληροφορίες για το DBI θα βρείτε στο `<http://www.fugue.com/dbi/>`
- Κεντρικό ftp site `<ftp://ftp.demon.co.uk/pub/perl/db>`
- Διάφορα DBI link `<http://www-ccs.cs.umass.edu/db.html>`
- Διάφορα DBI link `<http://www.odmg.org/odmg93/updates_dbarry.html>`
- Διάφορα DBI link `<http://www.jcc.com/sql_stnd.html>`
- Η βάση PostgreSQL `<http://www.postgresql.org>`

ΑΠΑΙΤΗΣΕΙΣ:

- - να χτίσετε, να τεστάρετε και να εγκαταστήσετε την Perl 5 (τουλάχιστον την 5.002)
- - να χτίσετε, να τεστάρετε και να εγκαταστήσετε το module DBI (τουλάχιστον το 0.89)
- - να χτίσετε, να τεστάρετε και να εγκαταστήσετε την PostgreSQL (τουλάχιστον την 6.2)

11.2.3 Τεχνική υποστήριξη για το DBI

Παρακαλώ στείλτε τα σχόλιά σας και αναφορές λαθών στο

- `E.Mergl@bawue.de`

Συμπεριλάβετε παρακαλώ την έξοδο της εντολή perl -v, και perl -V, την έκδοση της PostgreSQL, την έκδοση του DBD-Pg, και την έκδοση του DBI στην αναφορά λαθών.

11.2.4 Τι είναι τα DBI, DBperl, Oraperl και *perl;

Όπως είπε και ο Tim Bunce, ο σχεδιαστής και συγγραφέας του DBI:

"Το DBI είναι ένα προγραμματιστικό interface (API) σε Perl για διασύνδεση με διασύνδεση με βάσεις δεδομένων. Οι προσδιορισμοί του DBI API ορίζουν ένα σύνολο από ρουτίνες, μεταβλητές και συμβάσεις που παρέχουν ένα συνεκτικό interface βάσης δεδομένων ανεξάρτητο από την πραγματική βάση που χρησιμοποιείται."

Με απλά λόγια, το DBI interface επιτρέπει στους χρήστες να έχουν διάφανη πρόσβαση σε πολλαπλούς τύπους βάσεων. Έτσι, αν συνδέεστε την Oracle, Informix, mSQL, Sybase ή οποιαδήποτε άλλη βάση, δε χρειάζεται να γνωρίζετε τους υποκείμενους μηχανισμούς του επιπέδου 3GL. Το API που ορίζεται από το DBI θα δουλέψει σε όλους τους τύπους βάσεων.

Ένα παρόμοιο πλεονέκτημα είναι η ικανότητα να συνδεθείτε σε δύο βάσεις διαφορετικού τύπου μέσω του ίδιου perl script, πχ, θέλετε να διαβάσετε δεδομένα από μια βάση Oracle και να τα εισάγετε σε μια βάση Informix μέσα στο ίδιο πρόγραμμα. Το επίπεδο DBI σας επιτρέπει να το κάνετε αυτό απλά και ισχυρά.

DBperl είναι το παλιό όνομα για τις προδιαγραφές του interface. Χρησιμοποιείται συνήθως για να υποδηλώσει τα perl4 modules για διασύνδεση με βάσεις, όπως, oraperl, isqlperl, ingperl και λοιπά. Αυτά τα interfaces δεν είχαν στάνταρ API και γενικά δεν υποστηρίζονται.

Να μια λίστα από DBperl modules, το αντίστοιχο DBI ταίρι τους και πληροφορίες υποστήριξης. Ερωτήσεις σχετικά με τον οδηγό DBI θα πρέπει να απευθύνονται στη λίστα αλληλογραφίας dbi-users .

```
\textgreek{'Onoma} Module \textgreek{Apaito'umenh} \textgreek{B'ash}     \textgreek{Suggraf'eac}
------------ -----------------   ----------          ---
Sybperl     Sybase              Michael Peppler     DBD::Sybase
                                <mpeppler@datamig.com>
                                http://www.mbay.net/~mpeppler
Oraperl     Oracle 6 & 7        Kevin Stock         DBD::Oracle
                                <dbi-users@fugue.com>
Ingperl     Ingres              Tim Bunce &         DBD::Ingres
                                Ted Lemon
                                <dbi-users@fugue.com>
Interperl   Interbase           Buzz Moschetti      DBD::Interbase
                                <buzz@bear.com>
Uniperl     Unify 5.0           Rick Wargo          -
                                <rickers@coe.drexel.edu>
Pgperl      Postgres            Igor Metz           DBD::Pg
                                <metz@iam.unibe.ch>
Btreeperl   NDBM                John Conover        SDBM;
                                <john@johncon.com>
Ctreeperl   C-Tree              John Conover        -
                                <john@johncon.com>
Cisamperl   Informix C-ISAM     Mathias Koerber     -
                                <mathias@unicorn.swi.com.sg>
Duaperl     X.500 Directory     Eric Douglas        -
            User Agent
```

Παρόλα αυτά, μερικά DBI modules έχουν επίπεδα προσομοίωσης DBperl. Έτσι, η DBD::Oracle έρχεται με ένα επίπεδο προσομοίωσης Oraperl, που σας επιτρέπει να τρέξετε κληρονομικά oraperl scripts χωρίς αλλαγές. Το επίπεδο προσομοίωσης μεταφράζει τις κλήσεις του oraperl API σε κλήσεις DBI και τις εκτελεί μέσω του DBI switch.

Να ένας πίνακας με πληροφορίες για τα επίπεδα προσομοίωσης :

```
Module          \textgreek{Ep'ipedo} \textgreek{prosomo'iwshc}      \textgreek{Kat'astash}
------          ---------------------     ------
DBD::Oracle     Oraperl                   \textgreek{Oloklhrwm'eno}
DBD::Informix   Isqlperl                  \textgreek{Up'o} \textgreek{an'aptuxh}
DBD::Sybase     Sybperl                   \textgreek{Doule'uei}; (\textgreek{qrei'azetai} \textgreek{e
DBD::mSQL       Msqlperl                  \textgreek{Ekd'ojhke} \textgreek{peiramatik'a} \textgreek{me
                                          DBD::mSQL-0.61
```

Η προσομοίωση Msqlperl είναι μια ειδική περίπτωση. Το Msqlperl είναι ένας οδηγός perl5 για mSQL βάσεις, αλλά δε συμμορφώνεται με τις προδιαγραφές του DBI. Η χρήση αποδοκιμάζεται ρητά για χάρη του DBD::mSQL. Μπορείτε να κατεβάσετε την Msqlperl από τη CPAN από το:

- <http://www.perl.com/cgi-bin/cpan_mod?module=Msqlperl>

11.2.5 Προδιαγραφές DBI

Υπάρχουν λίγες πηγές πληροφορίας για DBI.

- Προδιαγραφές DBI `<http://www.hermetica.com/technologia/perl/DBI/doc/dbispec>`

Υπάρχουν δύο προδιαγραφές διαθέσιμες σε αυτό το link, οι νέες Πρόχειρες (Draft) DBI προδιαγραφές, αυτό το κείμενο αλλάζει καθημερινά καθώς η ομάδα ανάπτυξης βαδίζει ορος ένα σταθερό interface, και οι παλιές ιστορικές DBperl προδιαγραφές από τις οποίες αναπτύχθηκε το τρέχον DBI interface.

Το τελευταίο κείμενο πρέπει να διαβαστεί μόνο και μόνο από ιστορικό ενδιαφέρον και να μη χρησιμοποιηθεί ως εγχειρίδιο προγραμματισμού, ή ως έγκυρο με οποιαδήποτε έννοια. Παρόλα αυτά, παραμένει μια ακόμη χρήσιμη πηγή αναφοράς.

Τεκμηρίωση POD Τα PODs είναι κομμάτια τεκμηρίωσης που ενσωματώνονται συνήθως σε προγράμματα perl που τεκμηριώνουν τον κώδικα “in place”, παρέχοντας ένα χρήσιμο βοήθημα για προγραμματιστές και χρήστες modules. Το POD για DBI και drivers γίνεται όλο και περισσότερο κοινότοπο, και η τεκμηρίωση για αυτά τα modules μπορεί να διαβαστεί με τις παρακάτω εντολές.

Προδιαγραφές DBI Τα POD για προδιαγραφές DBI μπορεί να διαβαστεί με την εντολή:

perldoc DBI

Oraperl Οι χρήστες του επιπέδου προσομοίωσης Oraperl που έρχεται μαζί με την DBD::Oracle, μπορούν να συνεχίσουν να διαβάσουν για το πως να προγραμματίσουν με το Oraperl πληκτρολογώντας:

perldoc Oraperl

Αυτό θα παράγει ένα ενημερωμένο αντίγραφο της αρχικής σελίδας βοηθείας oraperl, που γράφτηκε από τον Kevin Stock για perl4. Το API της oraperl εμφανίζεται και περιγράφεται εκεί πλήρως.

DBD::mSQL Οι χρήστες του DBD::mSQL module μπορούν να διαβάσουν για μερικές private functions και ιδιομορφίες αυτού του οδηγού (driver) πληκτρολογώντας:

perldoc DBD::mSQL

Συχνά Διατυπωμένες Ερωτήσεις (FAQ)

Το κείμενο, Frequently Asked Questions είναι επίσης διαθέσιμο ως τεκμηρίωση σε POD ! Μπορείτε να το διαβάσετε στο δικό σας σύστημα, πληκτρολογώντας:

perldoc DBI::FAQ

Αυτό μπορεί να είναι βολικότερο για ανθρώπους που δεν είναι συνδεδεμένοι μόνιμα ή άνετα, στο Internet.

Το POD γενικά Πληροφόρηση για τη συγγραφή POD, και για την φιλοσοφία του POD γενικότερα, μπορεί να διαβαστεί γράφοντας:

perldoc perlpod

Χρήστες που έχουν εγκαταστήσει το Tk module ίσως να ενδιαφέρονται να μάθουν πως υπάρχει διαθέσιμος ένας POD αναγνώστης βασισμένος σε Tk που λέγεται tkpod, και διαμορφώνει POD σε μια βολική και αναγνώσιμη μορφή.

“Περιπλανήσεις”, “Μεζεδάκια” και Παρατηρήσεις Υπάρχει μια σειρά από περιστασιακές “περιπλανήσεις” από διάφορους στις λίστες αλληλογραφίας για DBI.

- `<http://www.hermetica.com/technologia/perl/DBI/tidbits>`

“DBI – The perl5 Database Interface” Αυτό είναι ένα άρθρο από τον Alligator Descartes και τον Tim Bunce για τη δομή του DBI. Δημοσιεύτηκε στο τεύχος 5 του “The Perl Journal”. Είναι υπερβολικά καλό. Πηγαίνετε και αγοράστε το περιοδικό. Εδώ που τα λέμε, αγοράστε όλα τα τεύχη! “The Perl Journal”s WWW site is:

- `<http://www.tpj.com>`

"DBperl" Αυτό το άρθρο, το οποίο δημοσιεύτηκε στην έκδοση Νοεμβρίου του 1996 "Dr. Dobbs Journal" ασχολείται με DBperl.

"The Perl5 Database Interface" Αυτό είναι ένα βιβλίο γραμμένο από τον Alligator Descartes και εκδόθηκε από τους O'Reilly και Associates.

Λίστες Αλληλογραφίας υπάρχουν τρεις λίστες αλληλογραφίας για DBI που διαχειρίζεται ο Ted Lemon. Σε όλες μπορείτε να γραφτείτε και να ξεγραφτείτε από το World Wide Web στη διεύθυνση :

- Λίστες αλληλογραφίας `<http://www.fugue.com/dbi>`

Οι λίστες στις οποίες μπορούν να συμμετέχουν οι χρήστες είναι:

dbi-announce Αυτή η λίστα αλληλογραφίας μόνο για ανακοινώσεις. Αν δεν μπορείτε να χρησιμοποιήσετε την φόρμα της παραπάνω WWW σελίδας, εγγραφείτε στη λίστα με τον ακόλουθο τρόπο:

- Email: `dbi-announce-request@fugue.com`

με κείμενο στο σώμα (body) του μηνύματος 'subscribe'

dbi-dev Η λίστα αυτή στοχεύει σε χρήση από τους προγραμματιστές που κάνουν ανάπτυξη, που συζητάνε ιδέες και έννοιες για τα των DBI interface, API και driver. Αν δεν μπορείτε να χρησιμοποιήσετε την φόρμα της παραπάνω WWW σελίδας, εγγραφείτε στη λίστα με τον ακόλουθο τρόπο:

- Email: `dbi-dev-request@fugue.com`

με κείμενο στο σώμα (body) του μηνύματος 'subscribe'

dbi-users Αυτή η λίστα αλληλογραφίας είναι μια λίστα γενικών συζητήσεων που χρησιμοποιείται για αναφορές σφαλμάτων, συζήτηση προβλημάτων και γενικές αναζητήσεις. Αν δεν μπορείτε να χρησιμοποιήσετε την φόρμα της παραπάνω WWW σελίδας, εγγραφείτε στη λίστα με τον ακόλουθο τρόπο:

- Email: `dbi-users-request@fugue.com`

με κείμενο στο σώμα (body) του μηνύματος 'subscribe'

Χρονικά/Αρχεία των Λιστών Αλληλογραφίας

- Αμερικάνικα Αρχεία Λιστών Αλληλογραφίας `<http://outside.organic.com/mail-archives/dbi-users/>`
- Ευρωπαϊκά Αρχεία Λιστών Αλληλογραφίας `<http://www.rosat.mpe-garching.mpg.de/mailing-lists/PerlDB-Interest>`

11.2.6 Προβλήματα Μεταγλώττισης ή αλλιώς "Κόπηκε στις Εξετάσεις!"

Αν έχετε core dump, δοκιμάστε το module Devel::CoreStack για να δημιουργήσετε ένα stack trace από το core dump. Το Devel::CoreStack βρίσκεται στο CPAN στο:

- `<http://www.perl.com/cgi-bin/cpan_mod?module=Devel::CoreStack>`

Στείλτε Email στη λίστα dbi-users το stack trace, την έκδοση των modules, της perl, τα τεστ που κάνατε, την έκδοση του λειτουργικού συστήματος και όποιες άλλες σχετικές πληροφορίες. Όσες περισσότερες πληροφορίες στείλετε, τόσο γρηγορότερα θα ανιχνεύσουν οι προγραμματιστές τα προβλήματα. Αν δε μας στείλτε τίποτα, τότε μην περιμένετε τίποτα και από εμάς.

–>

11.2.7 Υποστηρίζεται το DBI κάτω από πλατφόρμες Windows 95 / NT ;

Η μεταφορά των DBI και DBD::Oracle Win32 είναι τώρα ένα στάνταρ τμήμα του DBI, έτσι, αν κατεβάσετε έκδοση του DBI μεγαλύτερη του 0.81 θα πρέπει να δουλεύει μια χαρά. Μπορείτε να έχετε πρόσβαση στις βάσεις Microsoft Access και SQL-Server από το DBI μέσω ODBC. Μαζί με το DBI-0.79 (και μετέπειτα) δίνεται ένα πειραματικό DBI 'επίπεδο προσομοίωσης' για το module Win32::ODBC. Ονομάζεται DBI::W32ODBC. Θα χρειαστείτε το module Win32::ODBC.

- Win32 DBI `<http://www.hermetica.com/technologia/perl/DBI/win32>`
- Win32 ODBC `<http://www.roth.net>`

11.2.8 Τι είναι το DBM; Και γιατί θα πρέπει να χρησιμοποιώ το DBI στη θέση του;

Το UNIX ευλογήθηκε από την αρχή με απλές "βάσεις" βασισμένες σε επίπεδα αρχεία, γνωστές σαν σύστημα dbm. Το dbm σας επιτρέπει να αποθηκεύετε δεδομένα σε αρχεία, και να ανακτάτε τα δεδομένα γρήγορα Παρόλαυτά, αυτό έχει και σοβαρά μειονεκτήματα.

Κλείδωμα Αρχείων (File Locking)

Τα συστήματα dbm δεν παρείχαν ιδιαίτερα ισχυρές δυνατότητες για κλείδωμα αρχείων, ούτε καμία ικανότητα για τη διόρθωση προβλημάτων που προέκυπταν από ταυτόχρονες εγγραφές [στη βάση].

Αυθαίρετες Δομές Δεδομένων (Arbitrary Data Structures)

Τα συστήματα dbm επιτρέπουν μόνο μια σταθερή δομή δεδομένων: ζευγάρια κλειδιών-τιμών (key-value pairs). Αυτή η τιμή μπορεί να είναι ένα σύνθετο αντικείμενο, όπως ένα [C] struct, αλλά το κλειδί πρέπει να είναι μοναδικό. Αυτό ήταν μεγάλος περιορισμός στη χρησιμότητα των συστημάτων dbm.

Παρόλαυτά, τα συστήματα dbm ακόμα είναι χρήσιμα για χρήστες με απλά σύνολα δεδομένων και περιορισμένους πόρους, μια και είναι γρήγορα, δυνατά και εξαιρετικά καλά ελεγμένα. Τα modules της Perl για την πρόσβαση σε συστήματα dbm έχουν ενσωματωθεί τώρα στην διανομή της, μέσω του module AnyDBM_File.

Περιληπτικά, το DBM είναι μια ικανοποιητικότατη λύση για (κατά κανόνα) read-only βάσεις, ή μικρά και απλά σύνολα δεδομένων. Παρόλαυτά, για περισσότερο ισχυρά και επεκτάσιμα σύνολα δεδομένων, χωρίς ούτε καν να αναφερθούμε και σε αξιόπιστο transactional locking, συνιστούμε στους χρήστες να χρησιμοποιήσουν DBI.

11.2.9 Υποστηρίζεται < το τάδε χαρακτηριστικό > στο DBI;

Θεωρώντας ότι το χαρακτηριστικό που ζητήσατε είναι ένα μη-στάνταρ, εξαρτώμενο από κάποια συγκεκριμένη βάση, χαρακτηριστικό, τότε η απάντηση είναι όχι.

Το DBI ανταναχλά ένα γενικό API που θα δουλέψει για τις περισσότερες βάσεις, και δεν παρέχει καμία επιπλέον λειτουργικότητα για κάποια συγκεκριμένη βάση δεδομένων (database-specific functionality).

Παρόλαυτά, οι συγγραφείς των οδηγών μπορούν, αν το επιθυμούν, να συμπεριλάβουν hooks προς εξειδικευμένες λειτουργίες/χαρακτηριστικά κάποιας βάσης δεδομένων μέσω της μεθόδου func που ορίζεται στο DBI API. Όσοι αναπτύσσουν scripts θα πρέπει να σημειώσουν πως η λειτουργικότητα που παρέχεται μέσω των μεθόδων func είναι απίθανο να μεταφερθεί σε άλλες βάσεις.

11.2.10 Χρησιμεύει καθόλου το DBI για προγραμματισμό σε CGI;

Με μια λέξη, ναι! Το DBI είναι εξαιρετικά χρήσιμο για προγραμματισμό σε CGI! Στην πραγματικότητα, θα τολμούσα να πω ότι ο προγραμματισμός σε CGI είναι μία από τις κύριες χρήσεις του DBI.

Το DBI δίνει την ικανότητα στους προγραμματιστές CGI να ενισχύσουν τις βάσεις που έχουν WWW-fronted, πράγμα που παρέχει στους χρήστες τη δυνατότητα να παίξουν με τεράστιες ποσότητες δομημένων δεδομένων. Επίσης, στην περίπτωση που ένα site έχει περισσότερη κίνηση από ότι ο server της βάσης δεδομένων μπορεί να αντιμετωπίσει, το DBI δίνει τη δυνατότητα αναβάθμισης του server της βάσης δεδομένων στο παρασκήνιο, χωρίς αλλαγές στα cgi scripts.

11.2.11 Πώς πετυχαίνω ταχύτερους χρόνους σύνδεσης με DBD Oracle και CGI;

Ο Apache httpd διατηρεί μια δεξαμενή (pool) από παιδιά του (httpd children) για να εξυπηρετούν τις αιτήσεις των clients.

Χρησιμοποιώντας το module του Apache mod_perl του Doug MacEachern, ο μεταφραστής της perl (perl interpreter) ενσωματώνεται στα παιδιά του httpd. Τα CGI, DBI, και τα υπόλοιπα αγαπημένα σας modules μπορούν να φορτωθούν κατά την εκκίνηση κάθε παιδιού (child). Αυτά τα modules δε θα ξαναφορτωθούν εάν δεν αλλάξουν στο δίσκο.

Για περισσότερες πληροφορίες σχετικά με τον Apache, δείτε το Apache Project's WWW site:

- Apache Project WWW site `<http://www.apache.org>`
- Mod_perl module `<http://www.perl.com/cgi-bin/cpan_mod?module=mod_perl>`

11.2.12 Πώς επιτυγχάνω μόνιμες (persistent) συνδέσεις μεταξύ DBI και CGI

Χρησιμοποιώντας το module Apache::DBI module του Edmund Mergl, τα logins στη βάση δεδομένων αποθηκεύονται και αντιστοιχίζονται σε ένα httpd child. Αν η εφαρμογή σας βασίζεται σε έναν μόνο χρήστη της βάσης δεδομένων, η σύνδεση μπορεί να γίνεται με κάθε child. Μέχρι σήμερα, οι συνδέσεις στις βάσεις δεδομένων δεν μπορούν να μοιράζονται μεταξύ διαφορετικών παιδιών του httpd.

Μπορείτε να κατεβάσετε το Apache::DBI από τη CPAN μέσω:

- `<http://www.perl.com/cgi-bin/cpan_mod?module=Apache::DBI>`

11.2.13 "Όταν τρέχω ένα perl script από την γραμμή εντολής, δουλεύει, αλλά όταν το τρέχω από τον httpd, αποτυγχάνει!" Γιατί;

Βασικά, ένας λόγος για τον οποίο μπορεί να συμβαίνει αυτό είναι ότι ο χρήστης από τον οποίο τρέξατε το script στη γραμμή εντολών έχει σωστά ρυθμισμένα τις μεταβλητές του περιβάλλοντος, στην περίπτωση του DBD::Oracle, μεταβλητές όπως $ORACLE_HOME, $ORACLE_SID ή TWO_TASK.

Η διεργασία httpd τρέχει συνήθως κάτω από το χρήστη nobody, πράγμα που σημαίνει ότι δεν έχει ρυθμισμένο το περιβάλλον του. Οποιοδήποτε script επιχειρήσει να τρέξει κάτω από αυτήν την κατάσταση θα αποτύχει σίγουρα.

Για να λύσετε αυτό το πρόβλημα, θέστε το περιβάλλον για τη βάση σας σε ένα μπλοκ BEGIN ( ) στην αρχή του script σας. Έτσι θα διορθωθεί το πρόβλημα.

Παρόμοια, θα πρέπει να ελέγξετε το logfile με τα httpd λάθη για τυχόν ενδείξεις, όπως και τα "Idiot's Guide To Solving Perl / CGI Problems" και "Perl CGI Programming FAQ" για περισσότερες πληροφορίες. Είναι απίθανο το πρόβλημα να έχει σχέση με το DBI. Διαβάστε και τα δύο αυτά κείμενα προσεκτικά!

- Το "Idiot's Guide To Solving Perl / CGI Problems" θα το βρείτε στο: `<http://www.perl.com/perl/faq/index.html>`

11.2.14 Μπορώ να κάνω multi-threading με το DBI;

Μέχρι την ημερομηνία αυτού του FAQ, όχι. Η perl δεν υποστηρίζει multi-threading. Παρόλαυτά, το multi-threading αναμένεται να γίνει μέρος του πυρήνα της διανομής της perl στην έκδοση 5.005, που σημαίνει ότι το DBI θα μπορεί να υποστηρίζει multi-threading αρκετά σύντομα μετά απ' αυτό.

Για να δείτε παραδείγματα κώδικα OCI για την Oracle που έχει multi-threaded εντολές SELECT, πηγαίνετε στο:

- `<http://www.hermetica.com/technologia/oracle/oci/orathreads.tar.gz>`

11.2.15 Πώς μπορώ να καλέσω stored procedures με το DBI;

Αν υποθέσουμε ότι έχετε δημιουργήσει μια stored procedure μέσα σε κάποια βάση, πχ, στην Oracle, μπορείτε να χρησιμοποιήσετε το $dbh->do για να εκτελέσετε άμεσα τη διαδικασία (procedure). Για παράδειγμα,

$dbh->do("BEGIN someProcedure END");

11.2.16 Πώς μπορώ να πάρω τις τιμές που επιστρέφονται από τις stored procedures με το DBI;

Θυμηθείτε, όμως, να κάνετε έλεγχο για λάθη!

```
$sth = $dbh->prepare( "BEGIN foo(:1, :2, :3); END;" );
$sth->bind_param(1, $a);
$sth->bind_param_inout(2, \$path, 2000);
$sth->bind_param_inout(3, \$success, 2000);
$sth->execute;
```

11.2.17 Πώς μπορώ να δημιουργήσω ή να καταστρέψω μια βάση δεδομένων με το DBI;

Η δημιουργία και διαγραφή μιας βάσης είναι έννοιες πολύ αφηρημένες για να υποστηρίζονται ικανοποιητικά από το DBI. Για παράδειγμα, η Oracle δεν υποστηρίζει καν την έννοια της διαγραφής της βάσης! Στην Oracle επίσης, ο διακομιστής της βάσης δεδομένων (database server) είναι ουσιαστικά η βάση, ενώ στην mSQL, η διεργασία του διακομιστή (server process) τρέχει μια χαρά χωρίς να έχει δημιουργηθεί καμία βάση σε αυτόν. Το πρόβλημα είναι τόσο ανόμοιο για να το θίξουμε.

Γι' αυτό το λόγο, μερικοί οδηγοί υποστηρίζουν δημιουργία και διαγραφή βάσεων μέσω των private func methods. Δείτε την τεκμηρίωση των οδηγών που χρησιμοποιείτε για να δείτε αν υποστηρίζουν αυτό το μηχανισμό.

11.2.18 Πώς διαχειρίζεται το DBI τις τιμές NULL ;

Οι τιμές NULL στο DBI αντιμετωπίζονται όπως η τιμή undef. Οι τιμές NULL μπορούν καταχωρηθούν στις βάσεις ως NULL, για παράδειγμα:

```
$rv =
    $dbh->do( "INSERT INTO table VALUES( NULL )" );
```

αλλά όταν ερωτηθούν, οι τιμές NULL θα πρέπει να συγκριθούν με την undef. Αυτό είναι στάνταρ για όλους τους οδηγούς.

11.2.19 Τι είναι αυτές οι μέθοδοι func (func methods);

Η μέθοδος func ορίζεται στο DBI σαν το σημείο εισόδου για λειτουργικότητα εξαρτώμενη από τη βάση, πχ, η ικανότητα να δημιουργήσετε ή να διαγράψετε βάσεις. Το να καλέσετε αυτές τις μεθόδους που εξαρτώνται από τους οδηγούς είναι απλό, για παράδειγμα, για να καλέσετε τη μέθοδο createDatabase που παίρνει μια παράμετρο, θα γράφαμε:

```
$rv =
    $dbh->func( 'argument', 'createDatabase' );
```

Όσοι αναπτύσσουν λογισμικό θα πρέπει σημειώσουν ότι οι μέθοδοι func δεν είναι μεταφέρσιμες μεταξύ διαφορετικών βάσεων.

11.2.20 Εμπορική Υποστήριξη και Εκπαίδευση

Το Interface της Perl5 για βάσεις είναι ένα ΔΩΡΕΑΝ λογισμικό. ΕΡΧΕΤΑΙ ΧΩΡΙΣ ΚΑΜΙΑ ΕΓΓΥΗΣΗ ΟΠΟΙΟΥΔΗΠΟΤΕ ΕΙΔΟΥΣ.

Παρόλαυτά, μερικοί οργανισμοί παρέχουν είτε τεχνική υποστήριξη είτε προγράμματα εκμάθησης του DBI.

Perl Clinic : Η Perl Clinic μπορεί να κλείσει συμβόλαια για εμπορική υποστήριξη σε Perl, DBI, DBD::Oracle και Oraperl. Υποστήριξη παρέχεται από την εταιρεία στην οποία δουλεύει ο Tim Bunce, ο συγγραφέας του DBI. Για περισσότερες υπηρεσίες σχετικά με τις υπηρεσίες τους, δείτε το :

- `<http://www.perl.co.uk/tpc>`

12 Εργαλεία Διαχείρισης της PostgreSQL

12.1 PGACCESS - Ένα γραφικό εργαλείο για τη διαχείριση της PostgreSQL

Το PgAccess είναι ένα Tcl/Tk interface για την PostgreSQL. Συμπεριλαμβάνεται ήδη στη διανομή της PostgreSQL. Ρίξτε μια ματιά σε αυτό το web site για νεότερη έκδοση

- `<http://www.flex.ro/pgaccess>`
- Αν έχετε οποιοδήποτε σχόλιο, πρόταση για βελτίωση, μη διστάσετε να στείλετε e-mail στο : `teo@flex.ro`

Χαρακτηριστικά του PgAccess

Παράθυρα του PgAccess : Κεντρικό παράθυρο , δημιουργία πινάκων (table builder) , Εμφάνιση πινάκων (ερωτήσεων) (table(query) view), δημιουργία ερωτήσεων οπτικά (visual query builder).

Πίνακες

- - εμφάνιση πινάκων, μέγιστο 200 εγγραφές (αλλάζει από το μενού preferences)
- - αλλαγή μεγέθους στήλης, σέρνοντας την κάθετη γραμμή (καλύτερα στο χώρο του πίνακα από ότι στην επικεφαλίδα του πίνακα)
- - text wrap στα κελιά - αποθήκευση της εμφάνισης κάθε πίνακα
- - import/export σε εξωτερικά αρχεία (SDF,CSV)
- - δυνατότητες φιλτραρίσματος (δώστε ένα φίλτρο σαν price>3.14)

- - δυνατότητες ταξινόμησης (δώστε χειροκίνητα το(τα) πεδία προς ταξινόμηση)
- - editing in place
- - βελτιωμένο βοήθημα για παραγωγή πινάκων (table generator assistant)
- - βελτιωμένες δυνατότητες διόρθωσης πεδίων

Ερωτήσεις (Queries)

- - δημιουργεί, διορθώνει και αποθηκεύει "ερωτήσεις που καθορίζει ο χρήστης"
- - αποθηκεύει ερωτήσεις σαν views
- - εκτελεί ερωτήσεις
- - εμφάνιση αποτελεσμάτων ερωτήσεων τύπου select
- - διαγραφή και αλλαγή ονόματος ερώτησης
- - Γραφικό χτίσιμο ερώτησης με δυνατότητες drag & drop. Για όσους από εσάς είχατε στήσει το Tcl/Tk plugin για το Netscape Navigator, μπορείτε να το δείτε να δουλεύει πατώντας εδώ

Sequences

- - δημιουργία sequences, διαγραφή και έλεγχός τους Συναρτήσεις
- - δημιουργία, έλεγχος και διαγραφή συναρτήσεων σε γλώσσα SQL

Μελλοντικές υλοποιήσεις

- - σχεδιασμός πινάκων (προσθήκη νέων πεδίων, αλλαγή ονόματος, κλπ.)
- - καθορισμός συνάρτησης
- - δημιουργία report
- - βασικό scripting

Αν έχετε οποιοδήποτε σχόλιο, πρόταση για βελτίωση, μη διστάσετε να στείλετε e-mail στο :

- `teo@flex.ro`

Περισσότερες πληροφορίες για τη βιβλιοθήκη libgtcl

Θα χρειαστείτε επίσης να δηλώσετε προς χρήση τη βιβλιοθήκη διασύνδεσης PostgreSQL σε Tcl σαν ένα Tcl/Tk 'load'-able module. Ονομάζεται libpgtcl και ο πηγαίος κώδικάς της βρίσκεται στον κατάλογο της PostgreSQL /src/interfaces/libpgtcl. Θα χρειαστείτε ιδιαίτερα, μια βιβλιοθήκη libpgtcl που να μπορεί να φορτωθεί από Tcl/Tk. Αυτό είναι διαφορετικό τεχνικά από ένα τυπικό PostgreSQL loadable object αρχείο, επειδή η libpgtcl είναι μια συλλογή από object αρχεία. Κάτω από Linux, ονομάζεται libpgtcl.so. Μπορείτε να κατεβάσετε από εδώ μια έκδοση ήδη μεταγλωττισμένη για συστήματα Linux i386. Απλά αντιγράψτε το libpgtcl.so στον κατάλογο βιβλιοθηκών του συστήματος (/usr/lib)και ξεκινήστε. Μία λύση είναι να αφαιρέσετε από τον πηγαίο κώδικα τη γραμμή που περιέχει το φόρτωμα της libpgtcl.so και να φορτώσετε τη pgaccess.tcl όχι με wish, αλλά με pgwish (ή wishpg) αυτό το wish με το οποίο είχε γίνει link με τη βιβλιοθήκη libpgtcl.

12.2 Windows Interactive Query Tool για PostgreSQL (WISQL ή MPSQL)

Η MPSQL παρέχει στους χρήστες ένα γραφικό SQL interface για την PostgresSQL. Η MPSQL είναι παρόμοια με το SQL Worksheet της Oracle και το εργαλείο ερωτήσεων του Microsoft SQL Server WISQL. Έχει ωραίο γραφικό interface και ιστορικό εντολών. Επίσης μπορείτε να κάνετε cut και paste και έχει και άλλα ωραία χαρακτηριστικά που βελτιώνουν την παραγωγικότητα.

- `<http://www.troubador.com/~keidav/index.html>`
- Email: `keidav@whidbey.com`
- `<http://www.ucolick.org/~de/>` στο αρχείο tcl_syb/wisql.html
- `<http://www.troubador.com/~keidav/index.html>`
- Email: `de@ucolick.org`

12.3 Interactive Query Tool(ISQL) για την PostgreSQL - Ονομάζεται PSQL

Το ISQL είναι για τερματικά με Character command line. Περιλαμβάνεται στη διανομή, και λέγεται PSQL. Μοιάζει πολύ με το ISQL της Sybase, και το SQLplus της Oracle. Στο unix prompt δώστε την εντολή 'psql' η οποία θα σας βγάλει σε psql> prompt.

```
\textgreek{Pat'hste} \h \textgreek{gia} \textgreek{na} \textgreek{de'ite} \textgreek{th} \textgreek{bo'hje
```

Πολύ φιλικό στο χρήστη και εύκολο στη χρήση. Μπορείτε να έχετε πρόσβαση σε αυτό και από scripts του κέλυφους.

12.4 MPMGR - Εργαλείο Διαχείρισης Βάσεων για PostgresSQL

Το MPMGR παρέχει ένα γραφικό interface διαχείρισης της PostgresSQL. Θα το βρείτε στο

- `<www.mutinybaysoftware.com/>`
- Email: `keidav@mutinybaysoftware.com`
- `<http://www.troubador.com/~keidav/index.html>`
- Email: `keidav@whidbey.com`
- `<http://www.ucolick.org/~de>` in file tcl_syb/wisql.html
- WISQL για την PostgreSQL `<http://www.ucolick.org/~de/Tcl/pictures>`
- Email: `de@ucolick.org`

13 Ρυθμίζοντας πολλαπλά κουτιά PostgreSQL σε μια οθόνη

Μπορείτε να στοιβάζετε τα κουτιά-cpu και να συνδεθείτε σε μια οθόνη και να χρησιμοποιήσετε το διακόπτη για να διαλέξετε το host. Αυτό εξοικονομεί χώρο και δε χρειάζεστε πολλές οθόνες, πληκτρολόγια και ποντίκια (εξοικονομείτε περίπου 100 με 500 US δολάρια ανά σετ) και γλιτώνετε επίσης από την ακαταστασία.

Χρησιμοποιώντας αυτή την τεχνική, μπορείτε να στοιβάζετε πολλούς PostgreSQL servers (για ανάπτυξη, έλεγχο, παραγωγή), Web servers, ftp servers, Intranet servers, Mail servers, News servers σε ένα πολύ ψηλό ράφι. Ο διακόπτης μπορεί να χρησιμοποιηθεί επίσης για κουτιά με Windows 95/NT ή OS/2.

Ρίξτε μια ματιά στα παρακάτω sites:

- Τηλεφωνικές παραγγελίες από την DataComm Warehouse Inc, σε 24ωρη βάση, όλη την εβδομάδα, στο 1-800-328-2261. Παρέχουν κάθε τύπο υλικού Η/Υ `<http://www.warehouse.com>`. Το 4-θυρο KVM switch (PS/2) κοστίζει γύρω στα $89.99. Part No. DDS1354
- Network Technologies Inc `<http://www.networktechinc.com/servswt.html>` (120 δολάρια/PC 8 ports) το οποίο έχει λίστες για 'Server Switches' και 'switches μόνο για Video '
- Scene Double Inc, Αγγλία `<http://www.scene.demon.co.uk/qswitch.htm>`
- Cybex corporation `<http://www.cybex.com>`
- Raritan Inc `<http://www.raritan.com>`
- RealStar Solutions Inc `<http://www.real-star.com/kvm.htm>`
- Belkin Inc `<http://www.belkin.com>`
- Better Box Communications Ltd. `<http://www.betterbox.com/info.html>`
- Πηγαίνετε στο πλησιέστερο μαγαζί hardware και ζητήστε το "Server Switch" γνωστό και σαν "KVM Auto Switches".

Ψάξτε στη μηχανή αναζήτησης yahoo για να βρείτε περισσότερες εταιρείες με "Server Switches" ή "KVM Switches".

Συνίσταται ιδιαίτερα να έχετε ένα unix box ειδικά αφιερωμένο για κάθε PostgreSQL data-server για καλύτερη απόδοση. Καμία άλλη εφαρμογή/διεργασία δεν πρέπει να τρέχει σε αυτό το μηχάνημα. Δείτε το τμήμα "Επαγγελματικά" στις τοπικές εφημερίδες για τοπικούς πωλητές που πωλούν μόνο κουτιά intel, 13" μονόχρωμες οθόνες (οθόνη πολύ χαμηλού κόστους). Οι τοπικοί πωλητές πωλούν μόνο το hardware **χωρίς** Microsoft Windows/DOS (έτσι εξοικονομείτε $ 150). Δε χρειάζεστε έγχρωμη οθόνη για το διακομιστή της βάσης δεδομένων, καθώς μπορείτε να κάνετε απομακρυσμένη διαχείριση (remote administration) από έγχρωμο σταθμό εργασίας. Πάρτε το RedHat (ή κάποια άλλη διανομή) Linux cdrom από το -

- Linux System Labs Web site: `<http://www.lsl.com/>` 7 (U.S. δολάρια)
- Cheap Bytes Inc Web site: `<http://www.cheapbytes.com/>` 7 (U.S. δολάρια)

Σιγουρευτείτε ότι το υλικό που αγοράσατε υποστηρίζεται από το Redhat Linux. Δείτε το ftp site ης Redhat πριν αγοράσετε υλικό, για προτεινόμενους SCSI adapters, κάρτες βίντεο κλπ. Με $ 600 μόνο μπορείτε να πάρετε ένα ισχυρό κουτί intel με Redhat Linux που τρέχει PostgreSQL. Χρησιμοποιήστε odbc/jdbc/perl/tcl για να συνδεθείτε με την PostgreSQL από Windows95, OS/2, Unix Motif ή κάποιο web browser (π.χ. Redbaron, Opera, Netscape, 20 άλλοι). (οι Web browsers έχουν γίνει πια standard client).

Με αυτή την ιδέα μπορείτε να ελέγχετε πολλά κουτιά cpu με μια οθόνη και ένα πληκτρολόγιο!

Παρακάτω θα βρείτε ένα **απόσπασμα από την networktechnic Inc** για "Server Switches"

Παραγγείλετε τώρα στο 800-742-8324 (χωρίς δασμούς για Αμερική)

Για να λάβετε τον κατάλογό μας παρακαλώ στείλτε Email στη διεύθυνση: sales@networktechinc.com

ΧΕΙΡΙΣΤΕΙΤΕ ΠΕΡΙΣΣΟΤΕΡΑ ΑΠΟ ΕΝΑ PC με ένα πληκτρολόγιο, οθόνη και ποντίκι Οι ηλεκτρονικοί διακόπτες επιτρέπουν με ένα πληκτρολόγιο, οθόνη και ποντίκι να χειρισθείτε 64 PC. Ενσωματωμένοι μικροϋπολογιστές προσομοιώνουν την παρουσία πληκτρολογίου, οθόνης και ποντικιού με όλα τα προσαρτημένα PC 100% του χρόνου.

Χαρακτηριστικά και Εφαρμογές

- Πληκτρολόγιο, χειριστήριο πάνω στη συσκευή ή τηλεχειριστήριο

- 9 pin D Serial ή 6 pin miniDIN υποστήριξη για ποντίκι
- 5 pin DIN ή 6 pin miniDIN υποστήριξη πληκτρολογίου
- 1600x1200 ανάλυση οθόνης
- Χρησιμοποιεί κοινά καλώδια
- Σε Desktop κόψιμο ή και για rack

Προδιαγραφές - Πληκτρολόγιο

- Όλα τα βύσματα είναι θηλυκά
- 5 pin DIN ή 6 pin miniDIN
- Μπορεί να εκκινεί οποιαδήποτε στιγμή οποιονδήποτε Η/Υ

Χειριστήριο - Μπροστινά Κουμπιά

- Συνδεθείτε σε έναν Η/Υ με το πάτημα ενός κουμπιού
- Κρατήστε πατημένο οποιοδήποτε κουμπί για περισσότερο από μισό δευτερόλεπτο και μπείτε σε τρόπο λειτουργίας SCAN, BROADCAST ή COMMAND
- Τα LEDs υποδεικνύουν τον τρόπο λειτουργίας του πληκτρολογίου
- Πατήστε CTRL+* και μπείτε σε τρόπο λειτουργίας COMMAND
- Είναι διαθέσιμοι οι SCAN, BROADCAST και COMMAND τρόποι λειτουργίας

Ποντίκι

* 9 pin D serial

- ο διακόπτης NTI που προσομοιώνει το σειριακό ποντίκι της Microsoft
- ο 9 pin D αρσενικό για ποντίκι
- ο 9 pin D θηλυκό για τους Η/Υ

* 6 pin miniDIN

- ο διακόπτης NTI που προσομοιώνει το ποντίκι IBM PS/2
- ο Όλα τα βύσματα είναι θηλυκά

Καλωδιακό χειριστήριο/τηλεκοντρόλ

- Προαιρετικό –θα πρέπει να αγοραστεί ξεχωριστά
- Κάνει τις δουλείες του "Χειριστηρίου - Μπροστινών Κουμπιών"

Οθόνη-VGA

- Εύρος 150 MHz
- 1600X1200 ανάλυση χωρίς παραμόρφωση
- 1/4λα τα βύσματα είναι θηλυκά

14 Εφαρμογές και Εργαλεία για την PostgreSQL

14.1 Γλώσσα 4ης γενιάς (4GL) PostgreSQL για εφαρμογές με web και βάση δεδομένων - Πλατφόρμα ανάπτυξης AppGEN

Το AppGEN μπορείτε να το κατεβάσετε από το

- `<http://www.man.ac.uk/~whaley/ag/appgen.html>`
- `<ftp://ftp.mcc.ac.uk/pub/linux/ALPHA/AppGEN>`.

Το AppGEN είναι μια υψηλού επιπέδου, τέταρτης γενιάς γλώσσα και πλατφόρμα ανάπτυξης εφαρμογών για την παραγωγή εφαρμογών βασισμένων στο World Wide Web (WWW). Αυτές οι εφαρμογές τυπικά χρησιμοποιούνται στο internet ή στο intranet ενός οργανισμού. Οι εφαρμογές του AppGEN υλοποιούνται ως C scripts που ακολουθούν τα πρότυπα του Common Gateway Interface (CGI), τα οποία υποστηρίζονται από τους περισσότερους Web Servers.

Για να χρησιμοποιήσετε το AppGEN θα χρειαστείτε τα ακόλουθα :-

Την PostgresSQL, σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων

Έναν web server που να υποστηρίζει CGI όπως ο HTTPD του NSCA

Έναν μεταγλωττιστή ansi C όπως ο GCC

Το AppGEN αποτελείται από τα ακόλουθα εκτελέσιμα για Unix (Linux) :-

- defgen, το οποίο παράγει μια βασική πρότυπη εφαρμογή (template application) από κάποιες λογικές δομές δεδομένων. Οι εφαρμογές μπορούν να εισάγουν, ενημερώνουν, διαγράφουν και ψάχνουν εγγραφές στην βάση δεδομένων ενώ παράλληλα διατηρούν αυτόματα και την ακεραιότητα των δεδομένων (referential integrity).
- appgen, ο μεταγλωττιστής AppGEN που μεταγλωττίζει κώδικα appgen σε εκτελέσιμο CGI C κώδικα και κείμενα HTML έτοιμα για χρήση σε έναν web server.
- dbf2sql, μια εφαρμογή για μετατροπή αρχείων συμβατών με dBase III .dbf σε εκτελέσιμα SQL scripts. Αυτό επιτρέπει σε δεδομένα που φυλάσσονται στα περισσότερα πακέτα βάσεων δεδομένων για DOS/Windows να μεταφερθούν σε έναν SQL server όπως η PostgresSQL.
- Επιπλέον, το AppGEN παρέχει μια συλλογή από κείμενα HTML , αρχεία GIF και Java applets που χρησιμοποιούνται από το σύστημα, και φυσικά, όπως κάθε καλό λογισμικό, συμπεριλαμβάνει και τον πλήρη πηγαίο κώδικα.

Μπορείτε να επικοινωνήσετε με τον συγγραφέα, Andrew Whaley, στο

- `andrew@arthur.smuht.nwest.nhs.uk`

14.2 WWW Web interface για PostgresSQL - DBENGINE

Το dbengine είναι ένα plug 'n play Web interface για την PostgreSQL που δημιουργήθηκε από τον Ingo Ciechowski. Βρίσκεται στο :

- `<http://www.cis-computer.com/dbengine/>`

Σχετικά με το DBENGINE : Το dbengine είναι ένα interface διασύνδεσης WWW και Postgres95 που παρέχει απλή πρόσβαση σε οποιαδήποτε βάση μέσα σε λίγα λεπτά.

Το PHP 3 δίνει μια γλώσσα που μοιάζει με την Perl στα κείμενά σας, αλλά όχι πραγματική Perl ενώ το AppGen και το wdb-p95 απαιτούν να δημιουργήσετε μερικά αρχεία ρυθμίσεων για κάθε μία από τις βάσεις σας – μοιάζει σαν να πρέπει πρώτα απ' όλα να μάθετε κάποιο είδος νέας γλώσσας πριν να ξεκινήσετε.

Αντίθετα με τα άλλα εργαλεία, δεν απαιτείται να μάθετε ειδικό προγραμματισμό ή κάποια scripting language για να ξεκινήσετε να δουλεύετε με το dbengine. Δεν υπάρχει επίσης κάποιο αρχείο ρυθμίσεων για κάθε βάση, έτσι δε χρειάζεται να εξοικειωθείτε με μια νέα δομή. Παρόλαυτά - σε περίπτωση που θέλετε να έχετε πρόσβαση σε όλα τα χαρακτηριστικά του dbengine θα ήταν καλή ιδέα να μάθετε τη γλώσσα Perl.

Ολο το σύστημα μπορεί να ρυθμιστεί μέσω απλών χειρισμών σε μια πρόσθετη βάση δεδομένων που περιέχει πληροφορίες σχετικά με το πώς να οπτικοποιήσετε τα δεδομένα σας. Μπορείτε επίσης να ορίσετε virtual Fields τα οποία υπολογίζονται αμέσως πριν να εμφανιστούν στην οθόνη.

Άδεια Το dbengine είναι δωρεάν λογισμικό κάτω από τους ίδιους όρους όπως και η Perl. Διαβάστε την άδειά της αν δεν είστε σίγουροι για τι μπορείτε ή δεν μπορείτε να κάνετε. Σε τελική ανάλυση, αυτή είναι μια ελαστικότερη και πιο ήπια έκδοση από την GNU άδεια – δεν επηρεάζει τη δουλειά σας αν θέλετε να "δανειστείτε" κάτι από το dbengine ή να συγκεντρώσετε κομμάτια από αυτό σαν μέρος ενός εμπορικού προϊόντος!

14.3 Apache Webserver Module για PostgreSQL - NeoSoft NeoWebScript

Ο Apache είναι ένας πολύ γνωστός Web Server. Ένα module διασύνδεσης PostgreSQL με τον Apache Webserver βρίσκεται στο -

- `<http://www.neosoft.com/neowebscript/>`

Το NeoWebScript είναι μια γλώσσα προγραμματισμού που επιτρέπει και απλά και σύνθετα προγράμματα να ενσωματωθούν σε αρχεία HTML.

Όταν ζητείται μια σελίδα HTML που περιέχει ενσωματωμένο κώδικα NeoWebScript, ο webserver που υποστηρίζει NeoWebScript εκτελεί το(α) ενσωματωμένα script(s), και παράγει μια web σελίδα με περιεχόμενο κομμένο και ραμμένο στις προτιμήσεις του προγραμματιστή, η οποία δημιουργήθηκε με αυτό το πρόγραμμα.

Το NeoWebScript είναι ένας γρήγορος, ασφαλής, εύκολος τρόπος για να μάθετε ισχυρό, server-based προγραμματισμό στον κώδικα HTML των web σελίδων. Με το NeoWebScript, είναι εύκολο να φτιάξετε μετρητές (counters), φόρμες για email, γκράφιτι στις σελίδες, guest books και να κρατάτε στοιχεία για τους επισκέπτες, ακόμη και για έναν αρχάριο προγραμματιστή. Δείτε πόσο καλά το NeoWebScript κρατά τα δικά του σε σύγκριση με την PERL και το JavaScript.

Αν θέλετε να εγκαταστήσετε το NeoWebScript στον webserver σας, ο Webmaster σας πρέπει να διαβάσει το Sysop FAQ για να ξεκινήσει. Η "Θεωρία των Λειτουργιών" (Theory of Operations) θα εξηγήσει πώς δουλεύει το NeoWebScript, ενώ η εγκατάσταση θα σας καθοδηγήσει στα διάφορα στάδιά της. Η διαχείριση ασχολείται με θέματα ρύθμισης και εκτέλεσης του server, διάφοροι έλεγχοι επαληθεύουν τη σωστή λειτουργία των NeoWebScript σας και η αποσφαλμάτωση ασχολείται με τα προβλήματα του server.

Δεν υπάρχει κανένα κόστος για τη χρήση του NeoWebScript-2.2 για τον ISP σας, το intranet σας ή το extranet σας. Θα δείτε την πλήρη άδεια χρήσης, όταν εγγραφείτε για να το κατεβάσετε, αλλά η ουσία είναι πως θα θέλαμε 99 σπαρταριστά δολάρια αν θέλετε να το ενσωματώσετε στα δικά σας προϊόντα ή να το χρησιμοποιήσετε σε έναν server εμπορικών συνδιαλλαγών (π.χ. SSL).

Το NeoWebScript είναι ένα module για τον Apache webserver που σας επιτρέπει την ενσωμάτωση της γλώσσας προγραμματισμού Tcl/Tk στις ιστοσελίδες σας σαν ένα εργαλείο scripting. Επινοήθηκε από τον Karl Lehenbauer, τον Τεχνικό Διευθυντή της NeoSoft, και τεκμηριώθηκε, εμπλουτίστηκε και επεκτάθηκε από τους προγραμματιστές και τεχνικούς συγγραφείς της NeoSoft.

Ο Apache webserver είναι ο πλέον δημοφιλής webserver στον κόσμο, χρησιμοποιούμενος στο 68 % των sites που εξετάστηκαν.

Η Tcl/Tk είναι η πανίσχυρη, δωρεάν, ανεξάρτητη πλατφόρμας γλώσσα scripting, που αναπτύχθηκε από τον Δρ. John Ousterhout. Παραθέτουμε τα λόγια του

"Η Tcl/Tk επιτρέπει στους προγραμματιστές λογισμικού να κάνουν την δουλειά δέκα φορές ταχύτερα από ότι χρησιμοποιώντας εργαλειοθήκες (toolkits) βασισμένες σε C ή C++. Είναι επίσης μια τέλεια γλώσσα ενσωμάτωσης ήδη υπαρχόντων εφαρμογών που τις κάνει πιο γραφικές και τις εξοπλίζει με Internet-ικές δυνατότητες".

Ο Karl Lehenbauer, Ιδρυτής και Προϊστάμενος του Τεχνικού Τμήματος της NeoSoft, αποτελεί μέρος της ανάπτυξης της Tcl/Tk από τα πρώτα της στάδια. Μαζί με τον Mark Diehkans, δημιούργησαν την Extended Tcl, γνωστή επίσης και ως TclX ή NeoSoft Tcl, ένα ισχυρό σύνολο επεκτάσεων στη γλώσσα. Πολλές από τις βασικές εντολές της Tcl προέκυψαν από την Extended Tcl, και μετά εισήχθησαν από τον Dr. Ousterhout στο βασικό πυρήνα της γλώσσας.

NeoSoft Inc., 1770 St. James Place, Suite 500, Houston, TX 77056 USA

14.4 Οι HEITML επεκτάσεις της HTML στον server και μια γλώσσα 4ης γενιάς για την PostgreSQL

Το εργαλείο heitml είναι ένας άλλος τρόπος για να επικοινωνεί η postgres με το world wide web. Για περισσότερες πληροφορίες επικοινωνήστε με τον

```
Helmut Emmelmann H.E.I. Informationssyteme GmbH
Wimpfenerstrasse 23 \textgreek{Thl}. 49-621-795141
68259 Mannheim Germany \textgreek{Fax}. 49-621-795161
```

- Στείλτε email στον K. Helmut Emmelmann στο emmel@h-e-i.de
- Η κύρια σελίδα του Heitml <http://www.heitml.com>
- Η δευτερεύουσα σελίδα του Heitml <http://www.h-e-i.deom>

Το heitml είναι επεκτάσεις της HTML στην πλευρά του server και ταυτόχρονα γλώσσα 4ης γενιάς. Μπορείτε να γράψετε εφαρμογές για το web σε στυλ HTML χρησιμοποιώντας HTML-οειδή tags.

Το heitml (προφέρετε "Χαϊ"-ΤιΕμΕλ) είναι επέκταση του HTML και μια πλήρως εξοπλισμένη γλώσσα 4ης γενιάς που επιτρέπει σε εφαρμογές Web να αλληλεπιδρούν με δεδομένα που φυλάσσονται σε SQL βάσεις, χωρίς να καταφεύγουν σε περίπλοκα CGI scripts.

Το heitml επεκτείνει το HTML στην πλευρά του server, μετατρέποντας δυναμικά αρχεία ".hei" σε HTML και είναι λοιπόν συμβατό με κάθε web browser. Χρησιμοποιεί την γνώριμη, εύχρηστη σύνταξη HTML και παρέχει πλούτο προκατασκευασμένων Tags και Βιβλιοθηκών για να λύσουν θέματα που μέχρι τώρα χρειάζονταν CGI. Όπως και το XML, το heitml παρέχει tags ορισμένα από τον χρήστη. Με το heitml τα ορισμένα από τον χρήστη tags μετατρέπονται σε HTML και στέλνονται στον browser.

Το heitml στοχεύει και στους σχεδιαστές HTML και στους επαγγελματίες προγραμματιστές. Οι σχεδιαστές HTML μπορούν να χρησιμοποιούν heitml Tags για να χτιστούν δυναμικές σελίδες web, να προσπελαύνουν SQL βάσεις, ή να δημιουργούν σύνθετες εφαρμογές web. Μετρητές, βάσεις με στοιχεία καταχώρησης, φόρμες αναζήτησης, φόρμες email , ιεραρχικά μενού, όλα μπορούν να κατασκευαστούν απλά χρησιμοποιώντας προκαθορισμένα HTML-οειδή Tags που βρίσκονται στις διάφορες Βιβλιοθήκες.

Για τους προγραμματιστές, το heitml ενσωματώνει μια πλήρη γλώσσα 4ης γενιάς σε HTML

```
(\textgreek{p}.\textgreek{q}. <if>, <while>, \textgreek{kai} <let> Tags),
```

συν ισχυρές δυνατότητες υπολογισμού εκφράσεων με ακεραίους, πραγματικούς, δυαδικούς, αλφαριθμητικούς, tuples τελεστές. Τα Tuples αντιμετωπίζονται όπως στις μοντέρνες αντικειμενοστραφείς γλώσσες και φυλάσσονται σε μεταβλητές σωρών (heap variable). Οι μεταβλητές της heitml, συμπεριλαμβανομένων και όλων των σύνθετων δομών δεδομένων, φυλάσσονται στο σωρό διατηρώντας τις αρχικές τους τιμές μεταξύ των σελίδων χρησιμοποιώντας το Session Mode. Είναι δυνατόν να ορίσετε τα δικά σας tags ή tags του περιβάλλοντος και ακόμα και να επανορίσετε τα HTML-tags.

Το heitml κάνει δυνατό το να

- - - αναπτυχθούν Web Sites με ένα δομημένο και σπονδυλωτό τρόπο, ώστε να μειωθεί δραστικά η καθυστέρηση λόγω συντήρησης.

- - - αναπτυχθούν έξυπνα και αλληλεπιδραστικά Web Sites, με περιεχόμενο που προσαρμόζεται δυναμικά στις ανάγκες του χρήστη.

- - - εμφανίσετε το περιεχόμενο SQL βάσεων χωρίς να χρειαστεί καθόλου προγραμματισμός εκτός από το να χρησιμοποιήσετε τη βιβλιοθήκη μας των προκαθορισμένων "dba" Tags.

- - - αναπτυχθούν εφαρμογές σύνθετων βάσεων δεδομένων και καταλόγων αγορών χρησιμοποιώντας τις Session Variables

Το heitml τρέχει σε Linux με οποιονδήποτε Web Server που χρησιμοποιεί CGI interface, και είναι ιδιαίτερα γρήγορο (αποφεύγοντας την καθυστέρηση λόγω CGI) με τον APACHE Web Server χρησιμοποιώντας το Apache API. Μέχρι τώρα υποστηρίζονται οι βάσεις MSQL (έκδοση 1 και 2), PostgreSQL (έκδοση 6), mysql, και η yard. Το heitml δουλεύει επίσης σε Linux, BSDi, Solaris και SunOS, καθώς και σε Windows NT με CGI και ISAPI και ODBC και Windows 95.

Το heitml (σε linux) είναι δωρεάν για ερευνητική, μη-εμπορική και ιδιωτική χρήση. Τα εμπορικά Web Sites πρέπει να πληρώσουν τέλη για την άδεια χρήσης. Η πλήρης λειτουργική έκδοση του heitml είναι διαθέσιμη να την κατεβάσετε για μια περίοδο δοκιμής δωρεάν. (Σημειώστε, παρόλαυτά, ότι κάθε Web σελίδα ".hei" που αναπτύσσετε θα εμφανίζει ένα μήνυμα που θα το χαρακτηρίζει σαν έκδοση για μη εμπορική χρήση. Μετά την εγγραφή, θα λάβετε ένα κλειδί για να εξαφανίσετε αυτό το μήνυμα χωρίς να χρειαστεί να επανεγκαταστήσετε το πρόγραμμα.)

Το heitml (προφέρεται "Χαϊ"-ΤιΕμΕλ) επεκτείνει και εμπλουτίζει σημαντικά τη λειτουργικότητα της HTML με τον ορισμό tags και χαρακτηριστικά πλήρες προγραμματισμού. Αυτό κάνει δυνατή την ύπαρξη εφαρμογών δυναμικού περιεχομένου και βάσεων δεδομένων μέσα στον κόσμο της HTML, χωρίς CGI και εξωτερικό scripting ή άλλες γλώσσες προγραμματισμού. Αυτό σημαίνει ότι εσείς, σαν συγγραφέας HTML, μπορείτε να προσθέσετε εφαρμογές στις web σελίδες σας, χρησιμοποιώντας απλά μερικά νέα tags χωρίς CGI και καθόλου προγραμματισμό. Από την άλλη, σαν προχωρημένος χρήστης ή προγραμματιστής μπορείτε να δημιουργήσετε ισχυρές προγραμματιστικές βιβλιοθήκες με tag. Αυτή η προσέγγιση κάνει το heitml κατάλληλο τόσο για τους νεόφερτους στην HTML όσο και για τους επαγγελματίες προγραμματιστές. Το heitml τρέχει στο web server και δημιουργεί δυναμικά HTML, έτσι ώστε το heitml να είναι συμβατό με τα στάνταρ του internet και με οποιοδήποτε web browser. Επιτρέπει πλήρη πρόσβαση σε βάσεις ενώ προστατεύει το χρήστη από την περιττή πολυπλοκότητα του CGI. Το heitml έχει αναπτυχθεί σύμφωνα με πρόσφατες έρευνες και σε συστήματα κατασκευής μεταγλωττιστών και δοσοληψιών (transaction).

Οι σελίδες heitml αναπτύσσονται όπως και οι σελίδες HTML, με έναν text editor ή έναν HTML editor, και τοποθετούνται στο web server με το γνωστό τρόπο. Παρόλαυτά, οι σελίδες μπορούν να περιέχουν τώρα δυναμικά heitml tags και βιβλιοθήκες tag για πρόσβαση σε βάσεις. Μπορείτε να χρησιμοποιήσετε αυτά τα tags για να έχετε πρόσβαση σε βάσεις, να δημιουργήσετε δυναμικό περιεχόμενο, να στείλετε emails, και να δημιουργήσετε ακόμη ισχυρές εφαρμογές όπως εγγραφή σε βάσεις και συστήματα αγορών.

Οι νεόφερτοι στην γλώσσα HTML και οι επαγγελματίες προγραμματιστές θα μείνουν έκπληκτοι με το πόσο γρήγορα και εύκολα μπορούν να σχεδιάσουν ενδιαφέρουσες εφαρμογές όπως το Interactive Guestbook χωρίς να χρειάζεται να μάθουν να γράφουν σύνθετα και πολύπλοκα CGI scripts, απλά και μόνο χρησιμοποιώντας τα εργαλεία που παρέχονται με τη dba βιβλιοθήκη μας.

Το heitml συνοδεύεται από ένα μεγάλο πλήθος tag βιβλιοθηκών, για να δημιουργήσετε guestbooks, εφαρμογές συντήρησης της βάσης, επεκτάσιμες φόρμες ερωτήσεων, ισχυρές φόρμες email ή να δομήστε το web site σας χρησιμοποιώντας ένα ιεραρχικό μενού. Αυτά τα εργαλεία είναι έτοιμα, απλά προσθέστε τα αντίστοιχα tags στο web site σας.

Σαν έμπειρος προγραμματιστής, μπορείτε να κάνετε πλήρη χρήση της heitml αρχιτεκτονικής δυναμικών εγγραφών: Το heitml δεν είναι απλά μια scripting γλώσσα με δυναμικούς τύπους, ισχυρή στον υπολογισμό εκφράσεων, αναδρομικές ρουτίνες και πέρασμα εκτεταμένων παραμέτρων, αλλά χαρακτηρίζεται επίσης από δυναμικές εγγραφές για την αυτόματη διαχείριση των δεδομένων ενός session ανεξαρτήτου μεγέθους.

14.5 America On-line AOL Web server για PostgreSQL

Ο δωρεάν εμπορικός webserver, AOLserver έκδοσης 2.3 υποστηρίζει συνδέσεις στη βάση PostgreSQL έκδοση 6.2.1 και μεγαλύτερη. Για περισσότερες πληροφορίες επισκεφτείτε το

- AOL Web Server `<http://www.aolserver.com>`

14.6 Problem/Project Tracking System Application Tool για PostgreSQL

Βρίσκεται στο :

- `<http://www.homeport.org/~shevett/pts/>`

14.7 Μετατροπή αρχείων dbase dbf σε PostgreSQL

Το πρόγραμμα dbf2msql δουλεύει άψογα με mSQL και PostgreSQL. Μπορείτε να το βρείτε στο

- `<ftp://ftp.nerosworld.com/pub/SQL/dbf2sql/>`
- `<ftp://ftp.postgresql.org/pub/incoming/dbf2pg-3.0.tar.gz>`

Το πρόγραμμα αυτό γράφτηκε από τον Maarten Boekhold, Faculty of Electrical Engineering TU Delft, NL Computer Architecture and Digital Technique section

- `M.Boekhold@et.tudelft.nl`

Μπορείτε να χρησιμοποιήσετε επίσης python για να διαβάσετε αρχεία dbf και να τα φορτώσετε σε μια βάση δεδομένων postgres.

- Δείτε το `<http://www.python.org>`

15 Εργαλείο για Σχεδιασμό/Υλοποίηση Web Βάσεων Δεδομένων για την PostgreSQL - EARP

- `<http://www.oswego.edu/Earp>`
- `<ftp://ftp.oswego.edu>` στον κατάλογο 'pub/unix/earp'.

15.1 Τι είναι το EARP ;

Το EARP είναι ένα εργαλείο για σχεδιασμό/υλοποίηση Web βάσεων, που κάθεται πάνω από το σύστημα βάσης δεδομένων PostgreSQL. Στη λειτουργικότητά του περιλαμβάνονται:

- Ένα σύστημα γραφικού σχεδιασμού.
- Ένα interface για sendmail. (μπορεί να χειριστεί τα εισερχόμενα και εξερχόμενα mail)
- Ένα αυξημένο μηχανισμό ασφαλείας.
- Έναν οδηγό cgi.

15.2 Υλοποίηση

Η κύρια υλοποίηση του EARP είναι ένα εκτελέσιμο CGI που τρέχει κάτω από το δαίμονα http και παρέχει πρόσβαση στο διακομιστή της βάσης δεδομένων. Όλα τα εργαλεία σχεδιασμού είναι χτισμένα μέσα στον οδηγό, ο σχεδιασμός δεν γίνεται παρά μόνο σε web. Τα ίδια τα εργαλεία απαιτούν ένα γραφικό browser, η συμβατότητα των αντικειμένων που σχεδιάζονται με τα εργαλεία είναι ανεξάρτητα του τρόπου υλοποίησης, βασισμένα στο σχεδιασμό ατομικών προτιμήσεων.

15.3 Τι χρειάζεστε για να τρέξετε το EARP

Το EARP πρέπει να τρέχει πιθανά σε πολλές πλατφόρμες με λίγο ή καθόλου μεταφορά. Οι γνωστές πλατφόρμες που τρέχει είναι οι εξής:

- Solaris 2.5
- Linux 1.2.13+
- GNU C++
- PostgreSQL (έκδοση 1.01 / 1.02)
- netsite server
- NCSA httpd

- GNU C++
- PostgreSQL (έκδοση 1.01 / 1.02)
- NCSA httpd
- Apache httpd

Η τρέχουσα έκδοση (1.3) του Earp σχεδιάστηκε πάνω στην έκδοση του libpq που έρχεται με την PostgreSQL v1.01/1.02. Αν χρησιμοποιείτε μια πιο πρόσφατη έκδοση της Postgres, να περιμένετε πως το πρόγραμμα θα χρειαστεί μερικές αλλαγές για να δουλέψει σωστά. Στην αναπτυσσόμενη έκδοση (Earp 2.0), η υποστήριξη libpq ενσωματώνεται σαν module.

15.4 Πώς δουλεύει ;

Ένα από τα κυριότερα χαρακτηριστικά του EARP είναι ότι χρησιμοποιεί Αντικειμενοστραφή μέθοδο για να παράγει html σελίδες που επικοινωνούν με τη βάση. Οι περισσότερες σελίδες αποτελούνται από διάφορα αντικείμενα (objects). Κάθε αντικείμενο παράγεται από κάποιου τύπου εργαλείο και αφού του δοθεί ένα όνομα, τα αντικείμενα συνδέονται μεταξύ τους σε μία αλληλουχία που μπορεί να καλεστεί από το εργαλείο της σελίδας (page tool). Τα αντικείμενα είναι μπορούν να χρησιμοποιηθούν από πολλαπλές σελίδες. Υπάρχουν βασικά εργαλεία για HTML, Ερωτήσεις (Querys), να παίρνετε τιμές από τις φόρμες, να φορμάρετε ερωτήσεις και αντικείμενα εισόδου, και να συνδέετε μεταξύ τους τα αντικείμενα σε άλλα αντικείμενα. Περισσότερα εξελιγμένα εργαλεία περιλαμβάνουν το εργαλείο για mail και το multithreaded εργαλείο ερωτήσεων.

Αλλο χαρακτηριστικό του EARP είναι η έξτρα ασφάλεια. Η πρόσβαση σε διάφορες περιοχές του συστήματος EARP μπορεί να περιοριστεί με διάφορους τρόπους. Για να ευκολύνει την έξτρα ασφάλεια, το EARP ελέγχει για κάθε σύνδεση στο σύστημα, και προσδιορίζει σε ποια ids και γκρουπ ανήκει ο συνδεδεμένος agent. Η πρόσβαση σε περιοχές καθορίζεται ξεχωριστά, και ο συνδυασμός αποφασίζει αν επιτρέπεται η πρόσβαση σε μια συγκεκριμένη περιοχή του Earp. Επιπλέον, ό,τι απαιτείται για να υλοποιήσετε τα χαρακτηριστικά ασφαλείας είναι ένας http server που υποστηρίζει βασικό (ή καλύτερα) user authentication.

15.5 Πού θα το βρω ;

Μπορείτε να κατεβάσετε το EARP με anonymous ftp από το

- <ftp://ftp.oswego.edu> στον κατάλογο 'pub/unix/earp'.

16 PHP Hypertext Preprocessor, μια scripting γλώσσα για την PostgreSQL η οποία ενσωματώνεται σε html και εκτελείται στο Server

Εργαλείο διασύνδεσης με το WWW

- <http://www.php.net>
- <http://www.vex.net/php>

Το παλιό όνομα είναι Professional Home Pages (PHP 3) και το νέο όνομα είναι PHP Hypertext Pre-Processor

- Υπάρχουν Mirror sites σε πολλές χώρες όπως το www.COUNTRYCODE.php.net
- <http://www.fe.de.php.net>
- <http://www.sk.php.net>
- <http://php.iquest.net/>

Για ερωτήσεις, στείλτε e-mail στο :

- rasmus@lerdorf.on.ca

Το PHP 3 είναι μια scripting γλώσσα για την PostgreSQL η οποία ενσωματώνεται σε html και εκτελείται στο Server. Σας επιτρέπει να γράφετε απλά scripts στα αρχεία .HTML παρόμοια με την JavaScript, μόνο που αντίθετα με την JavaScript, το PHP 3 δεν εξαρτάται από το browser. Το JavaScript είναι μια γλώσσα η οποία ενσωματώνεται σε html και τρέχει στο client-μέρος ενώ το PHP 3 εκτελείται στο Server. Το PHP 3 είναι παρόμοιο στην σύλληψη με το προϊόν της Netscape LiveWire Pro. Αν έχετε τα απαιτούμενα χρήματα, και τρέχετε τον εμπορικό Server της Netscape και έχετε ένα από τα υποστηριζόμενα λειτουργικά συστήματα, ίσως θα πρέπει να ρίξετε μια ματιά στο LiveWire Pro. Αν σας αρέσει το δωρεάν, ταχύτατα αναπτυσσόμενο λογισμικό που έρχεται με όλο τον πηγαίο κώδικα πιθανά να σας αρέσει το PHP 3.

16.1 Κύρια χαρακτηριστικά

Υποστήριξη στάνταρ CGI, FastCGI και Apache module. Σαν ένα στάνταρ πρόγραμμα CGI, το PHP 3 μπορεί να εγκατασταθεί σε οποιαδήποτε μηχανή Unix και να τρέξει σε οποιοδήποτε Unix web server. Με την υποστήριξη του νέου στάνταρ FastCGI, το PHP 3 έχει το πλεονέκτημα των βελτιώσεων στην ταχύτητα που οφείλονται σε αυτό το μηχανισμό. Σαν Apache module, το PHP 3 γίνεται εξαιρετικά ισχυρό και τρομακτικά γρήγορη εναλλακτική του προγραμματισμού σε CGI.

- Καταγραφή των προσβάσεων (Access Logging) Με τη δυνατότητα καταγραφής των προσβάσεων του PHP 3, οι χρήστες μπορούν να συντηρήσουν το δικό τους μετρητή και καταγραφέα προσβάσεων (hit counting and logging). Δε χρησιμοποιεί με κανένα τρόπο τα κεντρικά αρχεία πρόσβασης στο σύστημα, και παρέχει παρακολούθηση προσβάσεων σε πραγματικό χρόνο. Το Log Viewer Script παρέχει μια σύντομη περίληψη των προσβάσεων σε ένα σύνολο σελίδων για κάθε ξεχωριστό χρήστη. Επιπλέον, το πακέτο μπορεί να ρυθμιστεί ώστε να παράγει ένα footer σε κάθε σελίδα το οποίο δείχνει πληροφορίες σχετικά με την πρόσβαση. Δείτε το τέλος αυτής της σελίδας για ένα παράδειγμα σε αυτά.

- Έλεγχος προσβάσεων Μια ενσωματωμένη οθόνη, βασισμένη στο web, χειρίζεται τις ρυθμίσεις για τον έλεγχο των προσβάσεων. Είναι δυνατό να δημιουργήσετε κανόνες για όλες ή κάποιες από τις web σελίδες που έχει κάποιο συγκεκριμένο άτομο, το οποίο θέτει διάφορους περιορισμούς για το ποιος μπορεί να βλέπει αυτές τις σελίδες και πώς θα τις βλέπει. Οι σελίδες μπορεί να προστατεύονται με κωδικό, να είναι πλήρως απαγορευμένες, να είναι απενεργοποιημένη η καταγραφή (logging disabled) και ακόμη περισσότερα βασισμένα στο domain του client, browser,τη διεύθυνση e-mail ή ακόμη και το σχετικό κείμενο.

- Υποστήριξη για Postgres Η Postgres είναι ένα εξελιγμένο δωρεάν RDBMS. Το PHP 3 υποστηρίζει την ενσωμάτωση Postgres95 και PostgresSQL SQL ερωτήσεων κατευθείαν μέσα σε αρχεία .html.

- Υποστήριξη RFC-1867 Αποστολή/Ανέβασμα Αρχείων (File Upload) Το Upload αρχείων είναι ένα νέο χαρακτηριστικό στο Netscape 2.0. Επιτρέπει στους χρήστες να στείλουν (upload) τα αρχεία τους στο web server. Το PHP 3 παρέχει το πραγματικό Mime decoding για να το κάνει αυτό να δουλέψει και επίσης παρέχει το πλαίσιο για να κάνετε κάτι χρήσιμο με το αρχείο που στείλατε, μόλος το λάβει ο server.

- HTTP-based authentication control Το PHP 3 μπορεί να χρησιμοποιηθεί για να δημιουργήσει μηχανισμούς authentication επί παραγγελία, βασισμένους σε HTTP, για τον Apache web server.

- Μεταβλητές, Πίνακες (arrays), Συσχετιστικοί πίνακες (Associative Arrays) Το PHP 3 υποστηρίζει μεταβλητές τύπων (typed variables), πίνακες και ακόμη και Perl-οειδείς συσχετιστικούς πίνακες (associative arrays). Όλα αυτά μπορούν να περνούν από τη μία web σελίδα στην άλλη χρησιμοποιώντας τις μεθόδους των φορμών, GET ή POST.

- Conditionals, While Loops Το PHP 3 υποστηρίζει μια γλώσσα scripting με όλα τα χαρακτηριστικά της C. Μπορείτε να έχετε συνθήκες if/then/elseif/else/endif όπως και while loops και δηλώσεις switch/case για να περιγράφει τη λογική ροή του πώς θα εμφανίζεται η html σελίδα.

- Extended Regular Expressions Τα Regular expressions χρησιμοποιούνται πολύ για ταυτοποίηση (pattern matching), αντικαταστήσεις των pattern και γενικό χειρισμό αλφαριθμητικών. Το PHP 3 υποστηρίζει όλους τους κοινούς τελεστές κανονικών εκφράσεων.

- Raw HTTP Header Control Η δυνατότητα να κάνουμε τις web σελίδες να στέλνουν ειδικά τροποποιημένες επικεφαλίδες HTTP ανάλογα με κάποιες συνθήκες είναι απαραίτητο για σχεδιασμό web sites υψηλού επιπέδου. Συχνή χρήση αυτού, είναι η αποστολή μιας τοποθεσίας: URL header για ανακατεύθυνση του πελάτη που έκανε την κλήση προς άλλο URL. Μπορεί να χρησιμοποιηθεί επίσης για να απενεργοποιήσουμε το caching ή να επεξεργαστούμε την τελευταία επικεφαλίδα ενημέρωσης.
- Δημιουργία GIF εικόνας επί τόπου/στον αέρα (On-the-fly) Το PHP 3 υποστηρίζει τη GD βιβλιοθήκη εικόνων του Thomas Boutell η οποία σας δίνει τη δυνατότητα να δημιουργήσετε GIF εικόνες επί τόπου.
- Υποστήριξη ISP "Safe Mode" Το PHP 3 υποστηρίζει το μοναδικό "Safe Mode" το οποίο κάνει ασφαλές το να έχετε πολλούς χρήστες να τρέχουν PHP scripts στον ίδιο server.
- Είναι δωρεάν! Και το τελικό ουσιαστικό χαρακτηριστικό. Το πακέτο είναι εντελώς δωρεάν. Βρίσκεται κάτω από την άδεια GPL η οποία σας επιτρέπει να χρησιμοποιήσετε το λογισμικό για οποιοδήποτε σκοπό, εμπορικό ή οποιονδήποτε άλλο. Δείτε το κείμενο της δημόσιας άδειας GNU για περισσότερες λεπτομέρειες.

16.2 Αναγνώριση/Ευχαριστίες

* Μεγάλα κομμάτια αυτού του κώδικα αναπτύχθηκαν στο και για το Πανεπιστήμιο του Τορόντο. Πολλές ευχαριστίες στον Lee Oattes του Τμήματος Ανάπτυξης Δικτύων του πανεπιστημίου για τις συνεχείς εποικοδομητικές κριτικές του.

* Ο κώδικας υποστήριξης της PostgreSQL γράφτηκε από τον Adam Sussman

- `asussman@vidya.com`

* Αναρίθμητοι άλλοι, βοήθησαν στο να ελεγχθεί και να αποσφαλματωθεί το πακέτο.

16.3 PHP 3 - Σύντομο Ιστορικό

Το PHP ξεκίνησε την ζωή του ως cgi κώδικας επικάλυψης (cgi wrapper) γραμμένος σε Perl. Δεν προοριζόταν για τίποτα πέρα από προσωπική χρήση. Το όνομα του πρώτου αυτού πακέτου ήταν Personal Home Page Tools, και αργότερα έγινε Personal Home Page Construction Kit.

Γράφτηκε ένα εργαλείο για να ενσωματώνει εύκολα ερωτήσεις SQL σε web σελίδες. Ήταν βασικά ένα ακόμα CGI wrapper που μετέφραζε ερωτήσεις SQL και έκανε την δημιουργία φορμών και πινάκων από τις ερωτήσεις μια εύκολη υπόθεση. Το εργαλείο αυτό ονομάστηκε FI (Form Interpreter).

Το PHP/FI έκδοση 2.0 είναι ένα ενιαίο πρόγραμμα, εξ'ολοκλήρου ξαναγραμμένο, που αποτελεί τον συνδυασμό των δύο παραπάνω πακέτων. Έχει τώρα πια εξελιχθεί σε σημείο που είναι μια απλή προγραμματιστική γλώσσα ενσωματωμένη σε αρχεία HTML. Το αρχικό ακρωνύμιο, PHP, έμεινε. Δεν είναι καν αντιπροσωπευτικό πια. Το PHP/FI χρησιμοποιείται περισσότερο από ολόκληρα web sites σήμερα, παρά για μικρές εγκαταστάσεις Προσωπικών Σελίδων (Personal Home Page). Με όποιο όνομα, εξαλείφει την ανάγκη για τα πολυάριθμα μικρά Perl cgi προγράμματα, επιτρέποντας σας να βάλετε μικρά scripts απευθείας στα HTML αρχεία σας. Αυτό επιταχύνει την ολική ταχύτητα των web pages σας, μια που η καθυστέρηση του κλωνισμού (forking) της Perl σε πολλαπλά αντίτυπα έχει εξαλειφθεί. Κάνει επίσης ευκολότερη την διαχείριση μεγάλων web sites τοποθετώντας όλα τα συστατικά μιας web page σε ένα μόνο html αρχείο. Ενσωματώνοντας υποστήριξη για διάφορες βάσεις, κάνει την ανάπτυξη σελίδων-με-δυνατότητες-χρήσης-βάσης τετριμμένη. Πολλοί βρίσκουν την ενσωματωμένη φύση του προγράμματος πολύ ευκολότερη παρά το να προσπαθούν να δημιουργούν ξεχωριστά αρχεία HTML και CGI.

Κατά τη διάρκεια αυτής της τεκμηρίωσης, κάθε αναφορά στο PHP, FI ή στο PHP/FI μιλάει για το ίδιο πράγμα. Οι διαφορά μεταξύ του PHP και του FI είναι μόνο θέμα αντίληψης. Και τα δύο χτίζονται από τον κώδικα της ίδιας διανομής. Τώρα το PHP/FI μετονομάστηκε σε PHP 3.

16.4 Τι μπορώ να κάνω, λοιπόν, με το PHP/FI ;

Το πρώτο πράγμα που θα παρατηρήσετε αν τρέξετε μια σελίδα μέσω του PHP/FI είναι ότι προσθέτει ένα footer με πληροφορίες σχετικά με τον αριθμό προσβάσεων στη σελίδα σας (αν έχετε μεταγλωττίσει την καταγραφή προσβάσεων στο εκτελέσιμο). Αυτό είναι ένα πολύ μικρό δείγμα του τι μπορεί να κάνει για σας το PHP/FI. Παίζει τον εξίσου σημαντικό ρόλο του μεταφραστή cgi σε φόρμες, από εκεί βγαίνει και to FI μέρος του ονόματος. Για παράδειγμα, αν δημιουργήσετε μια φόρμα σε μία από τις web σελίδες σας, χρειάζεστε κάτι για να επεξεργαστεί την πληροφορία σε αυτή τη φόρμα. Ακόμη και αν απλά θέλετε να περάσετε πληροφορία σε μια άλλη σελίδα, θα πρέπει να χρησιμοποιήσετε ένα πρόγραμμα cgi για να το κάνετε αυτό. Το PHP/FI κάνει εξαιρετικά εύκολο το να πάρετε δεδομένα από μια φόρμα και να κάνετε κάτι με αυτά.

16.5 Ένα απλό παράδειγμα

Ας υποθέσουμε ότι έχετε μια φόρμα:

```
<FORM ACTION="/cgi-bin/php.cgi/~userid/display.html" METHOD=POST>
<INPUT TYPE="text" name="name">
<INPUT TYPE="text" name="age">
<INPUT TYPE="submit">
<FORM>
```

Το αρχείο display.html θα μπορούσε να περιέχει κάτι σαν το παρακάτω:

```
< ?echo "Hi $ name, you are $ age years old!<p>" >
```

Είναι τόσο απλό! Το PHP/FI δημιουργεί αυτόματα μια μεταβλητή για κάθε input field στη φόρμα σας. Μπορείτε τότε να χρησιμοποιήσετε αυτές τις μεταβλητές στο αρχείο ACTION URL.

Το επόμενο βήμα αφού έχετε καταλάβει πώς να χρησιμοποιείτε τις μεταβλητές είναι να ξεκινήσετε να παίζετε με μερικά tags λογικής ροής στις σελίδες σας. Για παράδειγμα, αν θέλατε να εμφανίσετε διαφορετικά μηνύματα βασισμένα σε κάτι που εισάγει ο χρήστης, θα χρησιμοποιούσατε τη λογική if/else. Στο παραπάνω παράδειγμα, μπορούμε να εμφανίσουμε διαφορετικά πράγματα με βάση την ηλικία που έδωσε ο χρήστης αλλάζοντας το αρχείο display.html ως εξής:

```
<?
    if($age>50);
        echo "Hi $name, you are ancient!<p>";
    elseif($age>30);
        echo "Hi $name, you are very old!<p>";
    else;
        echo "Hi $name.";
    endif;
>
```

Το PHP/FI παρέχει μία πολύ ισχυρή scripting γλώσσα που θα κάνει πολύ περισσότερα απ' ότι κάνουν τα παραπάνω απλά παραδείγματα. Δείτε το τμήμα στο "PHP/FI Script Language" για περισσότερες πληροφορίες.

Μπορείτε να καθορίσετε επίσης στο PHP/FI ποιος επιτρέπεται να έχει πρόσβαση στις σελίδες σας. Αυτό επιτυγχάνεται με μια ενσωματωμένη οθόνη ρύθμισης. Με αυτή θα μπορούσατε για παράδειγμα να καθορίσετε ότι μόνο τα άτομα από συγκεκριμένα domains θα μπορούν να δουν τις σελίδες σας, ή θα μπορούσατε να δημιουργήσετε ένα κανόνα που θα προστάτευε μερικές σελίδες σας με κωδικό. Δείτε το τμήμα "Έλεγχος προσβάσεων" για περισσότερες πληροφορίες.

Το PHP/FI είναι επίσης ικανό να δέχεται uploads αρχείων από οποιοδήποτε web browser συμβατό με RFC-1867. Αυτό το χαρακτηριστικό επιτρέπει σε ανθρώπους να ανεβάσουν και text και binary αρχεία. Με τον έλεγχο προσβάσεων του PHP/FI και τις λογικές συναρτήσεις, έχετε πλήρη έλεγχο πάνω στο ποιος επιτρέπεται να ανεβάζει (upload) αρχεία και τι μπορεί να κάνει με το αρχείο αφού το ανεβάσει. Δείτε το τμήμα "File Upload" για περισσότερες πληροφορίες.

Το PHP/FI υποστηρίζει το πακέτο βάσης δεδομένων PostgreSQL. Υποστηρίζει ενσωματωμένες SQL ερωτήσεις στα .HTML αρχεία σας. Δείτε το αντίστοιχο τμήμα στην "Υποστήριξη της PostgreSQL" για περισσότερες πληροφορίες.

Το PHP/FI υποστηρίζει επίσης και το πακέτο βάσης δεδομένων mysql. Υποστηρίζει ενσωματωμένες SQL ερωτήσεις στα .HTML αρχεία σας. Δείτε το αντίστοιχο τμήμα στην "Υποστήριξη της mysql" για περισσότερες πληροφορίες.

16.6 CGI Redirection

16.6.1 Σημειώσεις για τον Apache 1.0.x

Ενας καλός τρόπος να τρέξετε το PHP/FI είναι χρησιμοποιώντας ένα cgi redirection module με τον Apache server. Παρακαλώ σημειώστε ότι δε χρειάζεται να ανησυχείτε για modules ανακατεύθυνσης αν χρησιμοποιείτε την έκδοση του Apache module του PHP/FI. Υπάρχουν δύο από αυτά τα modules ανακατεύθυνσης διαθέσιμα. Ένα από αυτά αναπτύχθηκε από τον Dave Andersen

- `angio@aros.net`

και είναι διαθέσιμο στο

- `<ftp://ftp.aros.net/pub/util/apache/mod_cgi_redirect.c>`

και το άλλο έρχεται πακέτο με τον Apache και λέγεται mod_actions.c. Αυτά τα modules είναι εξαιρετικά παρόμοια. Διαφέρουν ελαφρά στη χρήση τους. Και τα δύο έχουν τεσταριστεί και δουλεύουν με το PHP/FI.

Δείτε την τεκμηρίωση του Apache για το πώς μπορείτε να προσθέσετε ένα module. Γενικά, προσθέτετε το όνομα του module σε ένα αρχείο που ονομάζεται Configuration. Η γραμμή που πρέπει να προσθέσετε αν θέλετε να χρησιμοποιήσετε το module mod_actions είναι:

Module action_module mod_actions.o

Αν χρησιμοποιείτε το module mod_cgi_redirect.c προσθέστε αυτή τη γραμμή:

Module cgi_redirect_module mod_cgi_redirect.o

Στη συνέχεια, μεταγλωττίστε το httpd και εγκαταστήστε το. Για να ρυθμίσετε το cgi redirection χρειάζεται είτε να δημιουργήσετε ένα νέο τύπο mime στο αρχείο σας mime.types είτε να χρησιμοποιήσετε την εντολή AddType στο αρχείο σας srm.conf για να προσθέσετε τον mime τύπο. Για να προσθέσετε τον τύπο mime θα πρέπει να έχετε κάτι σαν το παρακάτω:

```
application/x-httpd-php phtml
```

Αν χρησιμοποιείτε το module mod_actions.c, χρειάζεται να προσθέσετε την παρακάτω γραμμή στο αρχείο σας srm.conf :

```
Action application/x-httpd-php /cgi-bin/php.cgi
```

Αν χρησιμοποιείτε το mod_cgi_redirect.c θα πρέπει να προσθέσετε την παρακάτω γραμμή στο αρχείο srm.conf :

```
CgiRedirect application/x-httpd-php /cgi-bin/php.cgi
```

Μην χρησιμοποιήσετε ταυτόχρονα και τα δύο mod_actions.c και mod_cgi_redirect.c.

Αφού έχετε εγκαταστήσει ένα από τα cgi redirection modules και τα ρυθμίσετε σωστά, θα μπορείτε να προσδιορίσετε ότι θέλετε ένα αρχείο να γίνει parsed από το php/fi απλά αλλάζοντας το extension του αρχείου σε .phtml. Επιπλέον, αν προσθέσετε ένα index.phtml στη γραμμή DirectoryIndex του αρχείου σας srm.conf, τότε η top-level σελίδα στον κατάλογο θα γίνει αυτόματα parsed από το php αν το index αρχείο ονομάζεται index.phtml.

16.6.2 Netscape HTTPD

Μπορείτε να ανακατευθύνετε αυτόματα τις αιτήσεις για αρχεία με συγκεκριμένη κατάληξη αρχείου ώστε να τα χειριστεί το PHP/FI με τη χρήση του Netscape Server CGI Redirection module. Αυτό το module είναι διαθέσιμο στο "File Archives" στη σελίδα του PHP/FI. Το αρχείο README που υπάρχει μέσα στο πακέτο εξηγεί καθαρά πώς να το ρυθμίσετε για χρήση με το PHP/FI.

16.6.3 NCSA HTTPD

Ο NCSA δεν υποστηρίζει προς το παρών modules, και έτσι για να κάνουμε ανακατεύθυνση cgi με τον server αυτόν, θα χρειαστείτε να τροποποιήσετε τον κώδικα του server σας. Μια διόρθωση/μπάλωμα (patch) που το κάνει για τον NCSA 1.5 διατίθεται στα αρχεία του site του PHP/FI.

16.7 Τρέχοντας το PHP/FI από τη γραμμή εντολών

Αν χτίσετε τη CGI έκδοση του PHP/FI, μπορείτε να το χρησιμοποιήσετε από τη γραμμή εντολών απλά γράφοντας: php.cgi filename όπου το filename είναι το αρχείο που θέλετε να κάνετε parse. Μπορείτε να δημιουργήσετε επίσης standalone PHP/FI scripts γράφοντας στην πρώτη γραμμή του script σας το εξής:

```
#!/usr/local/bin/php.cgi -q
```

Το "-q" απαγορεύει την εκτύπωση των HTTP headers. Μπορείτε να αφήσετε έξω αυτή την επιλογή αν θέλετε.

17 Διασύνδεση με PostgreSQL μέσω Python

Ο Python είναι μία scripting γλώσσα, αντικειμενοστραφής, σε μορφή μεταφραστή (interpreted). Ενας απλός στη χρήση (εύκολη σύνταξη, απλές και προφανείς δηλώσεις) και έχει πολλές επεκτάσεις για την δημιουργία γραφικών διασυνδέσεων (GUIs), επικοινωνία με το WWW, κλπ. Ενας έξυπνος web browser (τύπου HotJava) βρίσκεται σε ανάπτυξη (Νοέμβριος. 1995) και αυτό θα ανοίξει στους προγραμματιστές πολλές πόρτες. Ο python είναι πνευματικής ιδιοκτησίας του Stichting S Mathematisch Centrum, Amsterdam. The Neverlands και είναι ελεύθερης αναδιανομής.

Υποστηρίζει δυναμικό φόρτωμα αντικειμένων, κλάσεων, modules, και εξαιρέσεων (exceptions). Προφανής είναι η προσθήκη διασυνδέσεων με νέες βιβλιοθήκες μέσω κώδικα C, πράγμα που κάνει τον Python εύκολα κατάλληλο για ιδιόμορφες ρυθμίσεις.

Ο Python είναι μια scripting γλώσσα υψηλού επιπέδου με X interface. Ο Python διανέμεται με τα Linux cdroms που περιλαμβάνουν τα περισσότερο στάνταρ Python modules, μαζί με modules για διασύνδεση με το σύνολο Tix widget για την Tk.

Το PyGreSQL είναι ένα python interface για την PostgreSQL. Ενσωματώνει την βιβλιοθήκη ερωτήσεων PostgreSQL για να επιτρέπει την εύκολη χρήση των πανίσχυρων δυνατοτήτων της PostgresSQL μέσα από Python scripts. Το PyGreSQL γράφτηκε από τον D'Arcy J.M. Cain και τον Pascal Andre.

- Νεο site της PyGreSQL `<http://www.druid.net/pygresql/>`
- Συντηρείται από τον D'Arcy at `<http://www.druid.net/~darcy/>`
- Το παλιό site είναι στο `<ftp://ftp.via.ecp.fr/pub/python/contrib/Database/PyGres95.README>`
- D'Arcy J.M. Cain `darcy@druid.net`
- Pascal Andre `andre@chimay.via.ecp.fr`
- Pascal Andre `andre@via.ecp.fr`

17.1 Πού μπορώ να βρω το PyGres ;

Τα home sites των διαφορετικών πακέτων είναι:

- Python `<ftp://ftp.python.org:/pub/www.python.org/1.5/python1.5b2.tar.gz>`
- PyGreSQL `<ftp://ftp.druid.net/pub/distrib/PyGreSQL-2.1.tgz>`
- Old site `<ftp://ftp.via.ecp.fr/pub/python/contrib/Database/PyGres95-1.0b.tar.gz>`

Θα πρέπει να ψάξετε να βρείτε κάποιο mirror site κοντά στην περιοχή σας. Αναφερθείτε στα πληροφοριακά αρχεία για το πού θα βρείτε αυτά τα sites. Το PyGres95 θα πρέπει να βρίσκεται στους καταλόγους contrib των sites των Python και PostgreSQL.

17.2 Πληροφορίες και υποστήριξη

Αν χρειάζεστε περισσότερες πληροφορίες για αυτά τα πακέτα, επισκεφτείτε παρακαλώ τα web sites τους:

- Python : `<http://www.python.org/>`
- PostgreSQL : `<http://epoch.cs.berkeley.edu:8000/postgres95/index.html>`
- PyGreSQL `<ftp://ftp.druid.net/pub/distrib/PyGreSQL-2.1.tgz>`
- Old site PyGreSQL : `<http://www.via.ecp.fr/via/products/pygres.html>`

Για υποστήριξη :

- Λίστα αλληλογραφίας για PyGreSQL. Μπορείτε να εγγραφείτε στέλνοντας μήνυμα στο `majordomo@vex.net` με την γραμμή "subscribe pygresql name@domain" στο σώμα του μηνύματος αντικαθιστώντας το "name@domain" με την δική σας ηλεκτρονική διεύθυνση αλληλογραφίας.
- Newsgroup for Python : newsgroup comp.lang.python
- PyGreSQL : επικοινωνήστε με τον Andre στο `andre@via.ecp.fr` για αναφορές σφαλμάτων, ιδεών, σχολείων

18 Επικοινωνία μεταξύ PostgreSQL και WWW - WDB-P95

18.1 Σχετικά με το wdb-p95

WDB-P95 - Ένα Web interface για βάσεις PostgreSQL από τον J. Douglas Dunlop Βρίσκεται στο

- Νέο WDB από τον J Rowe βρίσκεται στο `<http://www.lava.net/beowulf/programming/wdb>`
- Νέες εκδόσεις του WWW-WDB στο `<http://www.eol.ists.ca/~dunlop/wdb-p95/>`
- Για ερωτήσεις ή για εγγραφή στις Λίστες Αλληλογραφίας, επικοινωνήστε με τον `dunlop@eol.ists.ca`

Είναι μια τροποποιημένη έκδοση του wdb-1.3a2 που παρέχει μια σύνδεση στο WWW για την PostgreSQL. Αυτή η έκδοση απαιτεί επίσης ένα Browser ικανό να χειρίζεται πίνακες HTML για την έξοδο σε πίνακες (tabular output). Αυτό δεν απαιτείται από το αρχικό wdb και μπορεί να επαναφερθεί στην παλιά μορφή του σχετικά εύκολα.

Μπορείτε να δοκιμάσετε τα CASI Tape και Image Query. Μπορείτε να ρίξετε μια γρήγορη ματιά στο Form Definition File (FDF) που χρησιμοποίησα για να φτιάξω τα CASI Tape και Image Query επίσης, που περιλαμβάνει ένα JOIN 2 πινάκων.

Αυτή η έκδοση περιέχει όλα τα απαραίτητα αρχεία για την εγκατάσταση και εκτέλεση του WDB-P95 σαν διασύνδεση με τις PostgreSQL βάσεις σας. Η μεταφορά του συστήματος σε άλλη βάση θα πρέπει να είναι σχετικά εύκολη - υπό την προϋπόθεση πως υποστηρίζει στάνταρ SQL και έχει interface σε Perl.

18.2 Είναι απαραίτητο ο διακομιστής της PostgreSQL, το pgperl, και το httpd να είναι στον ίδιο host;

Οχι - ο διακομιστής της PostgreSQL διακομιστής δε χρειάζεται να είναι στο ίδιο host. Το WDB-P95 επειδή καλείται από τον http δαίμονα, πρέπει να είναι στο ίδιο host - και καθώς το WDB-P95 γράφτηκε για να χρησιμοποιεί το Pg.pm - το pgperl πρέπει να είναι στο ίδιο host επίσης. Το Pgperl γράφτηκε χρησιμοποιώντας τη βιβλιοθήκη libpq, έτσι μπορεί να έχει πρόσβαση σε οποιοδήποτε διακομιστή PostgreSQL οπουδήποτε στο δίκτυο, όπως οποιοσδήποτε άλλος πελάτης PostgreSQL. Όπως φαίνεται παρακάτω

(WWW Client (Netscape)) => (HTTP Server (NCSA's http) + WDB-P95 + pgperl + libpq)=> (PostgreSQL server)

Οι παρενθέσεις () αναπαριστούν μηχανήματα.

Κάθε μηχάνημα μπορεί να είναι διαφορετικού τύπου : NT, SUN, HP, ... αλλά χρειάζεστε τη βιβλιοθήκη διασύνδεσης libpq για τον τύπο μηχανήματος που σκοπεύετε να χρησιμοποιήσετε το WDB-P95, για να μεταγλωττίσετε το pgperl. (Το σύστημα σχεδιάστηκε για να χρησιμοποιεί HTML πίνακες γι' αυτό είναι προτιμότερος ένας πρόσφατος WWW client).

19 Προγραμματιστικές διασυνδέσεις για "C", "C++", ESQL/C και Δυαδικοί Τελεστές για την PostgreSQL

19.1 Διασύνδεση γλώσσας "C" με την PostgreSQL

Συμπεριλαμβάνεται στη διανομή και ονομάζεται 'libpq'. Είναι παρόμοιο με τα Oracle OCI, Sybase DB-lib και τις βιβλιοθήκες Informix CLI .

19.2 Interface γλώσσας "C++" για την PostgreSQL

Συμπεριλαμβάνεται στη διανομή και ονομάζεται 'libpq++'.

19.3 ESQL/C για PostgreSQL

Ο 'Ενσωματωμένος προμεταγλωττιστής C' για την PostgreSQL, ESQL/C είναι όπως το Oracle Pro*C, και το Informix ESQL/C. Το PostgreSQL ESQL/C είναι ένα προγραμματιστικό περιβάλλον διασύνδεσης (API) με SQL, που επιτρέπει στον προγραμματιστή C να δημιουργεί ειδικές εφαρμογές με δυνατότητες διαχείρισης της βάσης δεδομένων. Σας επιτρέπει να χρησιμοποιείτε μια γλώσσα τρίτης γενεάς με την οποία είσαστε εξοικειωμένος και να εκμεταλλεύεστε παράλληλα και τα πλεονεκτήματα της Δομημένης Γλώσσας Ερωτήσεων (SQL).

Το ESQL/C αποτελείται από τα παρακάτω κομμάτια λογισμικού:

- Τις βιβλιοθήκες ESQL/C με συναρτήσεις C που δίνουν πρόσβαση προς τον διακομιστή της βάσης δεδομένων (database server).
- Τα αρχεία επικεφαλίδων του ESQL/C που δίνουν τους ορισμούς για τις δομές δεδομένων, τις σταθερές, και μακροεντολές χρήσιμες για το πρόγραμμα ESQL/C.
- Ο προεπεξεργαστής ESQL/C, είναι ένας προεπεξεργαστής πηγαίου κώδικα που μετατρέπει ένα C αρχείο που περιέχει εντολές SQL σε ένα εκτελέσιμο αρχείο.

Βρίσκεται στο

- Η ESQL/C για την PostgreSQL περιλαμβάνεται στις διάφορες διανομές.
- Το κύριο site είναι `<ftp://ftp.lysator.liu.se/pub/linus>`
- Email : `linus@epact.se`

19.4 Δυαδικοί Τελεστές για την PostgreSQL

Οι δυαδικοί τελεστές γράφτηκαν από τον Nicolas Moldavsky

- `nico@overnet.com.ar`

Οι "C" συναρτήσεις που υλοποιούν τους δυαδικούς τελεστές (AND, OR, XOR, συμπλήρωμα ως προς 2) σε pgsql. Μπορείτε να τα κατεβάσετε με anonymous FTP από το

- `<ftp://ftp.overnet.com.ar/pub/utils/linux/bitpgsql.tgz>`

Περιλαμβάνεται Makefile για Linux.

20 Japanese Kanji Κώδικας για την PostgreSQL

Βρίσκεται στο παρακάτω site

- `<ftp://ftp.sra.co.jp/pub/cmd/postgres/>`

21 Μεταφορά της PostgreSQL σε Windows 95/Windows NT

Η μεταφορά στα Windows 95/Windows NT είναι σε εξέλιξη. Η μεταφορά γίνεται χρησιμοποιώντας το gcc και το gmake για Win NT/95. Για να μεταγλωττίσουμε τον πηγαίο κώδικα σε win32 χρησιμοποιούμε το πρόγραμμα gnu-win32. Το GNU gcc είναι διαθέσιμο για win32. Πηγαίνετε στο site -

- <http://www.cygnus.com/misc/gnu-win32>

και κατεβάστε το αρχείο cdk.exe (αυτο-αποσυμπιεζόμενο αρχείο για gnu-win32)

Η μεταφορά μπορεί επίσης να γίνει χρησιμοποιώντας το ακόλουθο εργαλείο "Unix-Emulator on NT" από το

- <http://www.softway.com>

22 Λίστες αλληλογραφίας

22.1 Αποκτήστε ένα δωρεάν λογαριασμό e-mail

Ελέγξτε τις δωρεάν προσφορές - αποκτήστε δωρεάν λογαριασμούς e-mail από τα

- Yahoo <http://www.yahoo.com> κλικάρετε στο e-mail
- Lycos <http://www.lycos.com> κλικάρετε στους νέους λογαριασμούς e-mail (new email accounts)
- hotmail <http://www.hotmail.com> κλικάρετε στους νέους λογαριασμούς e-mail (new email accounts)

Γραφτείτε στη λίστα αλληλογραφίας της PostgreSQL και το Yahoo σας δίνει την επιπλέον δυνατότητα να δημιουργήσετε ένα ξεχωριστό φάκελο για τα e-mails της PostgreSQL, έτσι ώστε τα να μην ανακατώνονται με τα άλλα προσωπικά σας e-mail. Επιλέξτε το μενού Email- > Options- > Filters και διαλέξτε ξεχωριστό φάκελο για email. Με αυτό το λογαριασμό e-mail μπορείτε να παίρνετε ή να στέλνετε mail από οπουδήποτε στον κόσμο εφόσον έχετε πρόσβαση σε web σελίδα.

Αν έχετε κάποιο άλλο e-mail, μπορείτε να χρησιμοποιήσετε την επιλογή "Mail Filters" για να λαμβάνετε αυτόματα τα mails της PostgreSQL σε ξεχωριστό φάκελο. Με αυτό τον τρόπο τα mails σας είναι καλύτερα οργανωμένα και αποφεύγετε και το ανακάτεμα τ ους.

22.2 Αγγλική Λίστα Αλληλογραφίας

Δείτε τις Λίστες Αλληλογραφίας στην κεντρική σελίδα στο web στο :

- <http://www.postgresql.org/>
- Ερωτήσεις μέσω Email στο: pgsql-questions@postgresql.org
- Developers pgsql-hackers@postgresql.org
- Port specific questions pgsql-ports@postgresql.org
- Documentation questions pgsql-docs@postgresql.org

Θα λάβετε απαντήσεις μέσω e-mail σε λιγότερο από μία μέρα!!

Μπορείτε να γραφτείτε επίσης σε λίστες αλληλογραφίας. Δείτε επίσης το παραπάνω τμήμα 'Αποκτήστε ένα δωρεάν λογαριασμό e-mail'. Για να γραφτείτε ή να ξεγραφτείτε από μια λίστα, στείλτε mail στα

- pgsql-questions-request@postgresql.org
- pgsql-hackers-request@postgresql.org
- pgsql-ports-request@postgresql.org
- pgsql-docs-request@postgresql.org

Το κυρίως τμήμα (body) του μηνύματος θα περιέχει μόνο την ακόλουθη γραμμή

subscribe

(ή)

unsubscribe

22.3 Αρχείο της Λίστας Αλληλογραφίας

Κρατείται αρχείο των λιστών αλληλογραφίας σε html μορφή στις παρακάτω διευθύνσεις -

- Διαθέσιμη λίστα με βάση τις διευθύνσεις μέσω MHonarc και WWW στο `<http://www.postgresql.org/mhonarc/pgsql-questions>`
- `<ftp://ftp.postgresql.org>` στον κατάλογο /pub/majordomo

22.4 Ισπανική Λίστα Αλληλογραφίας

Υπάρχει τώρα και μια "ανεπίσημη" λίστα για την postgreSQL στα ισπανικά. Δείτε επίσης το παραπάνω τμήμα 'Δωρεάν Λογαριασμός για να οργανώσετε τα e-mails της PostgreSQL' . Για να εγγραφείτε, πρέπει να στείλετε ένα μήνυμα στο:

- majordomo@tlali.iztacala.unam.mx

Το κυρίως τμήμα (body) του μηνύματος θα περιέχει μόνο την ακόλουθη γραμμή:

inscripcion pgsql-ayuda

23 Τεκμηρίωση και Βιβλιογραφία

23.1 Οδηγοί εκμάθησης και Εγχειρίδια

Τα παρακάτω συμπεριλαμβάνονται στη διανομή της PostgreSQL σε μορφή postscript και σε unix σελίδες βοηθείας (man-pages). Αν έχετε πρόσβαση στο internet, μπορείτε να βρείτε τα κείμενα που αναφέρονται παρακάτω στο `<http://www.postgresql.org/docs>`

- "Οδηγός Εγκατάστασης"
- "Οδηγός χρήσης" για την PostgreSQL
- "Οδηγός ανάπτυξης-υλοποίησης" με λεπτομέρειες για τις εσωτερικές λειτουργίες της PostgreSQL.
- Online εγχειρίδια.
- Online εγχειρίδια σε HTML μορφή.
- Επίσης, εγχειρίδια σε μορφή Postscript για εκτύπωση.

23.2 Online Τεκμηρίωση

- Λίστα και περιγραφή των προκαθορισμένων τύπων δεδομένων και τελεστών

  `\textgreek{E'inai} \textgreek{tm'hma} \textgreek{twn} \textgreek{entol'wn} PSQL \textgreek{sthn} \tex`

- Λίστα των υποστηριζόμενων SQL λέξεων κλειδιά (keywords)

  `\textgreek{Up'arqei} \textgreek{'ena} script \textgreek{ston} \textgreek{kat'alogo} /tools \textgreek`

- Λίστες των υποστηριζόμενων εντολών -

  `\textgreek{Qrhsimopoi'hste} \textgreek{thn} \textgreek{entol'h} psql \h`

- Βασικές έννοιες σχεσιακής βάσης δεδομένων για PostgreSQL (υλοποίηση) και αρκετά online παραδείγματα (ερωτήσεις) -

  ```
  \textgreek{Koit'axte} \textgreek{sta} regression tests \textgreek{sto} src/test. \textgreek{Eke'i} \t
  regress/sql \textgreek{kai} suite/*.sql.
  ```

- Οδηγός εκμάθησης για την PostgreSQL.

  `\textgreek{Ja} \textgreek{bre'ite} SQL scripts \textgreek{gia} \textgreek{ekm'ajhsh} \textgreek{thc}`

  Δείτε επίσης το "Οδηγός εκμάθησης SQL για αρχάριους" στο Appendix B αυτού του κειμένου B ()

23.3 Χρήσιμη Βιβλιογραφία Αναφοράς

- "Understanding the New SQL: A Complete Guide" - από τους Jim Melton και Alan R.Simon

  `\textgreek{To} Morgan Kaufman Publisher \textgreek{e'inai} \textgreek{'ena} \textgreek{ap'o} \textgre`

- "A Guide to THE SQL STANDARD" - από τον C.J.Date

  `\textgreek{H} \textgreek{etaire'ia} Addison-Wesley Publishing \textgreek{'eqei} \textgreek{ep'ishc} \`

- SQL - The Standard Handbook, Νοέμβριος 1992

  ```
  Stephen Cannan \textgreek{kai} Gerard Otten
  McGraw-Hill Book Company Europe , Berkshire, SL6 2QL, England
  ```

- SQL Instant Reference, 1993

  ```
  Martin Gruber, Technical Editor: Joe Celko
  SYBEX Inc.  2021 Challenger Drive Alameda, CA 94501
  ```

- C.J.Date, "An introduction to Database Systems" (6th Edition), Addison-Wesley, 1995, ISBN 0-201-82458-2

  ```
  \textgreek{Aut'o} \textgreek{to} \textgreek{bibl'io} \textgreek{e'inai} \textgreek{h} \textgreek{B'ib
  \textgreek{Peri'eqei} \textgreek{plhrofor'iec} \textgreek{gia} \textgreek{kanonikopo'ihsh}, SQL, reco
  integrity, \textgreek{kai} extensions \textgreek{sto} \textgreek{arqik'o} \textgreek{sqesiak'o} \text
  \textgreek{'opwc} client/server \textgreek{sust'hmata} \textgreek{kai} \textgreek{Antikeimenostraf'h}
  \textgreek{anafor'ec} \textgreek{sumperilamb'anontai} \textgreek{gia} \textgreek{'osouc} \textgreek{j
  ```

- Stefan Stanczyk, "Theory και Practice of Relational Databases", UCL Press Ltd, 1990, ISBN 1-857-28232-9

  ```
  \textgreek{Peri'eqei} \textgreek{leptom'ereiec} \textgreek{me} \textgreek{jewr'ia} \textgreek{sqesiak
  \textgreek{kai} \textgreek{kanonikopo'ihsh}. \textgreek{All'a} \textgreek{den} \textgreek{kal'uptei}
  ```

- "The Practical SQL Handbook" Τρίτη Έκδοση, Addison Wesley Developers Press ISBN 0-201-44787-8

```
\textgreek{To} \textgreek{sunisto'ume} \textgreek{stouc} \textgreek{periss'oterouc} \textgreek{qr'hst
```

- Michael Stonebraker, "Readings in Database Systems", Morgan Kaufmann, 1988, ISBN 0-934613-65-6

```
\textgreek{To} \textgreek{bibl'io} \textgreek{e'inai} \textgreek{mia} \textgreek{sullog'h} \textgreek
```

- C.J.Date, "Relational Database - Selected Readings", Addison-Wesley, 1986, ISBN 0-201-14196-5

```
\textgreek{To} \textgreek{bibl'io} \textgreek{e'inai} \textgreek{mia} \textgreek{sullog'h} \textgreek
```

- Nick Ryan και Dan Smith, "Database Systems Engineering", International Thomson Computer Press, 1995, ISBN 1-85032-115-9

```
\textgreek{Aut'o} \textgreek{to} \textgreek{bibl'io} \textgreek{'eqei} \textgreek{leptom'ereiec} \tex
```

- Bipin C. Desai, "An introduction to Database Systems", West Publishing Co., 1990, ISBN 0-314-66771-7

```
\textgreek{Den} \textgreek{e'inai} \textgreek{gia} \textgreek{ton} \textgreek{apl'o} \textgreek{qr'hs
```

- Joe Celko "INSTANT SQL Programming"

```
Wrox Press Ltd.
Unit 16, 20 James Road, Tyseley
Birmingham, B11 2BA, England
1995
```

- Michael Gorman "Database Management Systems: Understanding και Applying Database"

```
Technology
QED \textgreek{kai} John Wiley
1991
```

- Michael Gorman "Enterprise Database for a Client/Server Environment" QED και John Wiley

```
\textgreek{Parousi'azei} \textgreek{tic} \textgreek{apait'hseic} \textgreek{gia} \textgreek{na} \text
\textgreek{m'esw} repository \textgreek{metamont'elwn}  \textgreek{kai} \textgreek{thc} \textgreek{qr
1993
```

Εκατοντάδες άλλοι τίτλοι είναι διαθέσιμοι για SQL ! Κοιτάξτε στα βιβλιοπωλεία.

23.4 Κείμενα προδιαγραφών ANSI/ISO SQL - SQL 1992, SQL 1998

Μπορείτε να βρείτε κείμενα προδιαγραφών ANSI/ISO SQL στα παρακάτω sites -

- <http://www.naiua.org/std-orgs.html>
- <http://www.ansi.org/docs> και κλικάρετε στο αρχείο cat_c.html και ψάξτε για "Database SQL"
- SQL92 standard <http://www.jcc.com> και κλικάρετε στο αρχείο sql_stnd.html
- ANSI/ISO SQL προδιαγραφές <http://www.contrib.andrew.cmu.edu/~shadow/sql.html> Εδώ θα βρείτε εγχειρίδιο για SQL.

23.5 Σύνταξη ANSI/ISO SQL 1992

Δείτε το Appendix A αυτού του κειμένου A ()

23.6 Σύνταξη ANSI/ISO SQL 1998

Οι προδιαγραφές SQL 1998 (SQL 3) είναι ακόμη υπό ανάπτυξη. Δείτε το 'Electronic Access to the SQL3 Working Draft' αυτού του κειμένου στο 23.9 ()

23.7 Οδηγός εκμάθησης SQL για αρχάριους

Δείτε το Appendix B αυτού του κειμένου B ()

23.8 Προσωρινή επέκταση (extension) στο SQL92

- Κείμενο για προσωρινή επέκταση SQL-92 <ftp://FTP.cs.arizona.edu/tsql/tsql2/>
- Προσωρινές προδιαγραφές SQL-3 <ftp://FTP.cs.arizona.edu/tsql/tsql2/sql3/>

Ο κατάλογος περιέχει τις προδιαγραφές της γλώσσας για την προσωρινή επέκταση στη στάνταρ γλώσσα SQL-92. Αυτή η νέα γλώσσα ονομάζεται TSQL2.

Οι προδιαγραφές της γλώσσας που παρουσιάζονται εδώ είναι η τελική έκδοση της γλώσσας.

Η αλληλογραφία μπορεί να σταλεί στον πρόεδρο του TSQL2 Language Design Committee, Richard T.Snodgrass, Department of Computer Science, University of Arizona, Tucson, AZ 85721,

- rts@cs.arizona.edu

Οι θυγατρικές και οι διευθύνσεις e-mail των μελών του TSQL2 Language Design Committee μπορούν να βρεθούν σε ξεχωριστό τμήμα στο τέλος των προδιαγραφών της γλώσσας.

Τα περιεχόμενα αυτού του καταλόγου είναι όπως παρουσιάζονται παρακάτω.

spec.dvi,.ps Προδιαγραφές Γλώσσας TSQL2, εκδόθηκε το Σεπτέμβριο του 1994

bookspec.ps Προδιαγραφές της Γλώσσας TSQL2, όπως εμφανίστηκαν στο βιβλίο TSQL2, που εκδόθηκε το Σεπτέμβριο του 1995 (δείτε παρακάτω).

sql3 αλλαγή των προτάσεων που υποβλήθηκαν στις επιτροπές ANSI και ISO SQL3.

Μαζί με τις προδιαγραφές της γλώσσας έχει συσχετιστεί μια συλλογή από σχόλια που αφορούν σε αποφάσεις σχεδιασμού, παροχή παραδειγμάτων, και απόψεις για το πώς μπορεί να υλοποιηθεί η γλώσσα. Αυτά τα σχόλια ήταν αρχικά προτάσεις στην επιτροπή του TSQL2 Language Design. Τώρα εξυπηρετούν ένα συγκεκριμένο σκοπό: να παρέχουν παραδείγματα του ότι έχει φτιαχτεί, ωθήσουν τις πολλές αποφάσεις που πάρθηκαν κατά το σχεδιασμό της γλώσσας, και να συγκρίνουν την TSQL2 με πολλές άλλες προτάσεις της γλώσσας που έχουν γίνει τα την τελευταία δεκαπενταετία. Θα πρέπει να δοθεί έμφαση ότι αυτά τα σχόλια δεν είναι τμήμα των προδιαγραφών της γλώσσας TSQL2, αλλά τη συμπληρώνουν και αναπτύσσουν με βάση αυτά. Οι κατάλληλες προδιαγραφές της γλώσσας είναι η τελευταία λέξη στην TSQL2.

Τα σχόλια, μαζί με τις προδιαγραφές της γλώσσας , αρκετοί δείκτες (indexes), και άλλο υλικό υποστήριξης, έχουν εκδοθεί σε ένα βιβλίο:

Snodgrass, R.T., editor, The TSQL2 Temporal Query Language, Kluwer Academic Publishers, 1995, 674+xxiv σελίδες.

Οι κριτικές εμφανίζονται στο βιβλίο σε μια συντομευμένη μορφή. Οι κριτικές παρέχονται πλήρεις σε αυτόν κατάλογο στο αρχείο eval.ps

Το αρχείο tl2tsql2.pl είναι ένα πρόγραμμα σε prolog που μεταφράζει την επιτρεπόμενη προσωρινή λογική σε TSQL2. Αυτό το πρόγραμμα γράφτηκε από τον Michael Boehlen

- boehlen@iesd.auc.dk

Μπορείτε να επικοινωνήσετε μαζί του για ένα paper που περιγράφει αυτή τη μετάφραση. Αυτό είναι μάλλον μια dated έκδοση του προγράμματος. Νεώτερες εκδόσεις είναι διαθέσιμες στο

- <http://www.cs.auc.dk/general/DBS/tdb/TimeCenter/Software>

(τα συστήματα TimeDB και Tiger).

23.9 Τμήμα 0 - Αποκτώντας τα κείμενα ISO/ANSI SQL

Αυτό το κείμενο σας δείχνει πώς να αποκτήσετε (νόμιμα) ένα αντίγραφο του προτύπου SQL-92 και του "τρέχοντος" SQL3 Working Draft.

Στο ANSI πρότυπο έχει πνευματικά δικαιώματα η ANSI, στο πρότυπο ISO η ISO.

Υπάρχουν δύο (2) τρέχοντα πρότυπα SQL, η δημοσίευση ANSI και η δημοσίευση ISO. Τα δύο αυτά πρότυπα είναι ολόιδια λέξη προς λέξη εκτός από τετριμμένα ζητήματα όπως ο τίτλος του κειμένου, οι επικεφαλίδες της σελίδας, η φράση "International Standard" αντί για "American Standard", και λοιπά.

Αγοράζοντας το πρότυπο SQL-92

Το πρότυπο ISO standard, ISO/IEC 9075:1992, Information Technology - Database Languages - SQL, είναι διαθέσιμο (Μάρτιος, 1993) στο ANSI :

```
American National Standards Institute
1430 Broadway
New York, NY 10018 (USA)
Phone (sales): +1.212.642.4900
```

και κοστίζει US$230.00. Η ANSI έκδοση, ANSI X3.135-1992, American National Standard για Συστήματα Πληροφοριών - Database Language SQL, δεν ήταν διαθέσιμο προς πώληση όταν γράφτηκε αυτό το κείμενο, αλλά αναμενόταν να είναι διαθέσιμο μεταξύ τέλη Μαρτίου και αρχές Μαΐου, 1993). Το κόστος του αναμένεται γύρω στα US$225.00.

Αν αγοράσετε κάποιο από τα κείμενα από το ANSI, θα έχετε έξτρα χρέωση για έξοδα μεταφοράς 7% επιπρόσθετα (δηλαδή, περίπου US$9.10). Για χώρες εξωτερικού τα έξοδα μεταφοράς θα είναι αναμφίβολα ακόμη περισσότερα. Το ANSI απαιτεί κάθε παραγγελία να συνοδεύεται από ένα χαρτί παραγγελίας της εταιρείας. Εναλλακτικά, μπορείτε να στείλετε ένα τσεκ αμερικάνικης τράπεζας ή σε αμερικάνικα δολάρια, τα οποία θα μετρηθούν πριν να σταλθεί η παραγγελία σας. (Υπάρχει μια εξαίρεση: αν η εταιρεία σας είναι ενσωματωμένο μέλος του ANSI, τότε το ANSI θα σας στείλει τα κείμενα και θα χρεώσει την εταιρεία σας.)

Το πρότυπο ISO είναι επίσης διαθέσιμο εκτός Αμερικής από εθνικά τοπικά τμήματα (τμήματα προτυποποίησης ανά χώρα) που είναι μέλη είτε του ISO (Διεθνής Οργανισμός Τυποποίησης - International Organization for Standardization) ή του IEC (Διεθνής Ηλεκτροτεχνική Επιτροπή - International Electrotechnical Commission).

Αντίγραφα της λίστα των εθνικών τμημάτων και οι διευθύνσεις τους είναι διαθέσιμες από το ANSI ή από άλλα εθνικά τμήματα. Είναι επίσης διαθέσιμα από το ISO:

```
International Organization for Standardization
Central Secretariat
1, rue de Varembi
CH-1211 Genhve 20
Switzerland
```

Αν προτιμάτε να παραγγείλετε το πρότυπο με πιο βολικό και γρήγορο τρόπο, θα πρέπει να πληρώσετε για να έχετε αυτό το προνόμιο. Μπορείτε να παραγγείλετε το ISO/IEC 9075:1992, Information Technology - Database Languages - SQL, από το:

```
Global Engineering Documents
2805 McGaw Ave
Irvine, CA 92714 (USA)
USA
\textgreek{Thl'efwno} (\textgreek{ap'o} \textgreek{opoiad'hpote} \textgreek{perioq'h}): +1.714.261.14
\textgreek{Thl'efwno} (\textgreek{m'ono} \textgreek{gia} \textgreek{Amerik'h}): (800)854-7179
```

και κοστίζει US$308.00. Δε ξέρω αν σε αυτή την τιμή συμπεριλαμβάνονται τα έξοδα αποστολής ή όχι, αλλά υποθέτω ότι η αποστολή στο εξωτερικό (τουλάχιστον) θα κοστίζει περισσότερο. Μπορούν να σας στείλουν το κείμενο αρκετά γρήγορα και δέχονται και τις "πιο γνωστές πιστωτικές κάρτες". Το Global δεν έχει ακόμη την ANSI έκδοση ούτε έχουν τιμή ή μια αναμενόμενη ημερομηνία (αν και την περιμένω ότι σε μερικές εβδομάδες μετά τη δημοσίευση του ANSI και με τιμή γύρω στα US$300.00).

Αγοράζοντας ένα αντίγραφο του SQL3 Working Draft

Μπορείτε να αγοράσετε μια εκτύπωση του SQL3 working draft από το ANSI X3 Secretariat, CBEMA (Computer και Business Equipment Manufacturers Association). Συνήθως διαθέτουν τις "πιο πρόσφατες εκδόσεις" του SQL3 working draft και τις πωλούν από US$60.00 μέχρι US$65.00. Μπορείτε να επικοινωνήσετε με το CBEMA στο:

```
CBEMA, X3 Secretariat
Attn: Lynn Barra
1250 Eye St.
Suite 200
Washington, DC 20005 (USA)
```

Μπορείτε να επικοινωνήσετε με τη Lynn Barra τηλεφωνικώς στο +1.202.626.5738 για να ζητήσετε ένα αντίγραφο, αν και το mail είναι προτιμότερο.

Ηλεκτρονική πρόσβαση στο SQL3 Working Draft

Η πιο πρόσφατη έκδοση (τη στιγμή της συγγραφής του κειμένου) του SQL3 (και για ANSI και ISO) working draft (και όλα τα τμήματά του) είναι διαθέσιμη με "anonymous ftp" ή με "ftpmail" από το:

```
gatekeeper.dec.com
```

```
\textgreek{sto} \textgreek{kat'alogo}
```

```
/pub/standards/sql/
```

Σε αυτόν τον κατάλογο υπάρχουν πολλά αρχεία. Υπάρχουν αρχεία PostScript και "plain text" (όχι απαραίτητα φορμαρισμένα, αλλά διαβάζονται σε μια οθόνη χωρίς ιδιαίτερο λογισμικό).

Γενικά, μπορείτε να βρείτε αρχεία με ονόματα όπως:

```
sql-bindings-mar94.ps
sql-bindings-mar94.txt
sql-cli-mar94.ps
sql-cli-mar94.txt
sql-foundation-mar94.ps
sql-foundation-mar94.txt
sql-framework-mar94.ps
```

```
sql-framework-mar94.txt
sql-psm-mar94.ps
sql-psm-mar94.txt
```

Καθώς παράγονται νέες εκδόσεις των κειμένων, το ”mar94” θα αλλάζει για να δείξει τη νέα ημερομηνία της δημοσίευσης (π.χ., ”aug94” είναι η αναμενόμενη ημερομηνία της επόμενης δημοσίευσης μετά το ”mar94”).

Επίσης, για τους αναγνώστες που δεν μπορούν να έχουν μια λίστα του καταλόγου με FTP, έχουμε τοποθετήσει στον ίδιο κατάλογο ένα αρχείο με το όνομα:

```
ls
```

Αυτό το αρχείο (έκπληξη!) περιέχει τη λίστα του καταλόγου.

Ανακτώντας Αρχεία κατευθείαν με χρήση του ftp

Αυτό είναι ένα παράδειγμα για το πώς να χρησιμοποιείτε το FTP. Συγκεκριμένα, δείχνει πώς να συνδεθείτε στο gatekeeper.dec.com, να πάτε στον κατάλογο όπου φυλάσσεται το βασικό κείμενο, και να μεταφέρετε το κείμενο στο μηχάνημά σας. Σημειώστε ότι το μηχάνημά σας πρέπει να έχει πρόσβαση στο Internet για να το κάνετε αυτό. Το login είναι ’ftp’ και ο κωδικός είναι η email διεύθυνσή σας (αυτό είναι γνωστό και σαν ανώνυμο ftp ’anonymous ftp’). Η εντολή ’type binary’ χρησιμοποιείται για να σιγουρέψετε ότι δε θα χάσετε κάποια bits από το(α) αρχείο(α) που θα λάβετε. Η ’get’ κατεβάζει ένα αρχείο τη φορά. Τα σχόλια στο παρακάτω script είναι μέσα σε angle brackets < σαν αυτά > .

```
% ftp gatekeeper.dec.com
Connected to gatekeeper.dec.com.
220- *** /etc/motd.ftp ***
     Gatekeeper.DEC.COM is an unsupported service of DEC Corporate Research.
     <...this goes on for a while...>
220 gatekeeper.dec.com FTP server (\textgreek{'ekdosh} 5.83 Sat ... 1992) ready.
Name (gatekeeper.dec.com:<yourlogin here>): ftp  <anonymous also works>
331 Guest login ok, send ident as password.
Password: <enter your email address here >
230 Guest login ok, access restrictions apply.
Remote system type is UNIX.  <or whatever>
Using binary mode to transfer files.
ftp> cd pub/standards/sql
250 CWD command successful.
ftp> dir
200 PORT command successful.
150 Opening ASCII mode data connection for /bin/ls.
total 9529
-r--r--r--  1 root     system     357782 Feb 25 10:18 x3h2-93-081.ps
-r--r--r--  1 root     system     158782 Feb 25 10:19 x3h2-93-081.txt
-r--r--r--  1 root     system     195202 Feb 25 10:20 x3h2-93-082.ps
-r--r--r--  1 root     system      90900 Feb 25 10:20 x3h2-93-082.txt
-r--r--r--  1 root     system    5856284 Feb 25 09:55 x3h2-93-091.ps
-r--r--r--  1 root     system    3043687 Feb 25 09:57 x3h2-93-091.txt
226 Transfer complete.
ftp> type binary
200 Type set to I.
ftp> get x3h2-93-082.txt
200 PORT command successful.
150 Opening BINARY mode data connection for x3h2-93-082.txt (90900 bytes).
226 Transfer complete.
90900 bytes received in 0.53 seconds (166.11 Kbytes/s)
```

```
ftp> quit
% <\textgreek{to} \textgreek{arqe'io} \textgreek{e'inai} \textgreek{t'wra} \textgreek{ston} \textgreek{k
```

Ανακτώντας Αρχεία χωρίς απευθείας υποστήριξη ftp

Η Digital Equipment Corporation, όπως και πολλές άλλες εταιρείες, παρέχει υπηρεσία ftp email. Η απάντηση μπορεί να χρειαστεί για να φτάσει αρκετές μέρες, αλλά παρέχει μια υπηρεσία ισοδύναμη με το ftp για αυτούς που δεν έχουν απευθείας ftp πρόσβαση στο Internet. Η διεύθυνση του server είναι:

ftpmail@decwrl.dec.com

Το παρακάτω script θα ανακτήσει το PostScript αρχείο της τελευταίας έκδοσης του SQL3 κειμένου:

```
reply joe.programmer@imaginary-corp.com
connect gatekeeper.dec.com anonymous
binary
compress
```

Το παρακάτω script θα ανακτήσει το PostScript αρχείο της τελευταίας έκδοσης του SQL3 κειμένου:

```
reply joe.programmer@imaginary-corp.com
connect gatekeeper.dec.com anonymous
binary
compress
uuencode
chdir /pub/standards/sql
get x3h2-93-091.ps
quit
```

Η πρώτη γραμμή από τις εντολές του script ζητά από το διακομιστή (server) να σας επιστρέψει τα ζητούμενα αρχεία. Θα πρέπει να αντικαταστήσετε το "joe.programmer@imaginary-corp.com" με την διεύθυνσή σας στο Internet. Το αρχείο σε αυτό το παράδειγμα, x3h2-93-091.ps, επιστρέφεται σε μορφή "compress"ed "uuencode"d σαν 34 διαφορετικά μηνύματα email. Αν το περιβάλλον σας δεν παρέχει εργαλεία για επανακατασκευή τέτοιων αρχείων, τότε μπορείτε να ανακτήσετε το αρχείο σαν plain text με το ακόλουθο script:

```
reply joe.programmer@imaginary-corp.com
connect gatekeeper.dec.com anonymous
chdir /pub/standards/sql
get x3h2-93-091.ps
quit
```

Αλλά σας προειδοποιούμε, το αρχείο .ps πολύ πιθανά θα σταλθεί σε περισσότερα από 70 τμήματα!

Για να ανακτήσετε οποιοδήποτε άλλο συγκεκριμένο αρχείο, εκτός του x3h2-93-091.ps, απλά αντικαταστήστε το "x3h2-93-091.ps" με το όνομα του αρχείου που επιθυμείτε. Για να πάρετε τη λίστα όλων των διαθέσιμων αρχείων του καταλόγου, αντικαταστήστε το "get x3h2-93-091.ps" με το "dir".

23.10 Τμήμα 1 - Τρέχουσα κατάσταση του ISO/ANSI SQL

Αυτό το κεφάλαιο είναι μια πηγή πληροφορίας σχετικά με την πρότυπη διεργασία SQL και την τρέχουσα κατάστασή της.

Τρέχουσα κατάσταση:

Η ανάπτυξή του είναι υπό εξέλιξη για να επεκτείνει την SQL σε μια ολοκληρωμένη υπολογιστικά γλώσσα για τον καθορισμό και διαχείριση persistent, σύνθετων αντικειμένων. Αυτό περιλαμβάνει : generalization και specialization hierarchies, πολλαπλή κληρονομικότητα, τύπους δεδομένων καθορισμένων από τον χρήστη, triggers και assertions, υποστήριξη για έμπειρα συστήματα, αναδρομικές ερωτήσεις, και επιπλέον εργαλεία διαχείρισης δεδομένων. Περιλαμβάνει επίσης τις προδιαγραφές των αφηρημένων τύπων δεδομένων (abstract data types -ADTs), object identifiers, μεθόδους, κληρονομικότητα, πολυμορφισμό, encapsulation, και όλες τις ευκολίες που σχετίζονται με τη διαχείριση δεδομένων των αντικειμένων.

Το φθινόπωρο του 1996, αρκετά τμήματα του SQL3 πέρασαν από ψηφοφορία ISO CD. Αυτά ήταν τα SQL/Framework, SQL/Foundation, και SQL/Bindings. Αυτή η ψηφοφορία απέτυχε (όπως ήταν αναμενόμενο) με 900 ή περίπου τόσα σχόλια. Στα τέλη Ιανουαρίου, έγινε κάποια συνάντηση σύνταξης του ISO DBL κατά την οποία επεξεργάστηκαν λύσεις για ένα μεγάλο αριθμό προβλημάτων τα οποία είτε ενσωματώθηκαν με σχόλια μέσω ψηφοφορίας είτε στάλθηκαν σαν ξεχωριστά papers. Μια και η συνάντηση σύνταξης του DBL δεν μπόρεσε να επεξεργαστεί όλα τα σχόλια, η συνάντηση σύνταξης παρατάθηκε. Η ολοκλήρωση της συνάντησης σύνταξης προγραμματίστηκε για τα τέλη Ιουλίου, 1997, στο Λονδίνο.

Μετά τη συνάντηση σύνταξης του Ιουλίου, αναμένεται να ζητηθεί μια τελική ψηφοφορία για όλα αυτά τα τμήματα της SQL. Η διαδικασία για την τελική ψηφοφορία θα κρατήσει 6 περίπου μήνες και θα γίνει μια συνάντηση σύνταξης DBL, μετά την οποία θα γίνει μια ψηφοφορία DIS και μια σχετικά γρήγορη ψηφοφορία IS .

Οι διαδικασίες ISO έχουν αλλάξει από το SQL/92, έτσι οι επιτροπές για την SQL δουλεύουν ακόμη για τον καθορισμό των ακριβών λεπτομερειών της διαδικασίας.

Αν πάνε όλα καλά, αυτά τα τμήματα της SQL3 θα γίνουν ένα επίσημο πρότυπο ISO/IEC στα τέλη του 1998, αλλά το πρόγραμμα είναι πολύ σφιχτό.

Το 1993, οι επιτροπές ανάπτυξης ANSI και ISO αποφάσισαν να χωρίσουν τη μελλοντική ανάπτυξη της SQL σε ένα πολυμερές πρότυπο. Τα τμήματα αυτά είναι:

- Τμήμα 1: Framework Μια μη τεχνική περιγραφή για το πώς είναι δομημένο το κείμενο.
- Τμήμα 2: Foundation Οι προδιαγραφές του συστήματος, συμπεριλαμβανομένου όλων των νέων χαρακτηριστικών ADT.
- Τμήμα 3: SQL/CLI To Call Level Interface.
- Τμήμα 4: SQL/PSM Οι προδιαγραφές των stored procedures, συμπεριλαμβανομένων υπολογιστικών συμπληρώσεων.
- Τμήμα 5: SQL/Bindings Τα Dynamic SQL και Embedded SQL bindings που πάρθηκαν από το SQL-92.
- Τμήμα 6: SQL/XA Οι προδιαγραφές SQL του δημοφιλούς XA Interface που αναπτύσσεται από την X/Open
- Τμήμα 7:SQL/Temporal Προσθέτει στα πρότυπα SQL ικανότητες σχετικές με το χρόνο.

Στην Αμερική, ολόκληρη η SQL3 επεξεργάζεται και σαν ANSI Domestic ("D") project και σαν ISO project. Το αναμενόμενο χρονικό παράθυρο για την ολοκλήρωση της SQL3 είναι το 1999.

Τα SQL/CLI και SQL/PSM επεξεργάζονται όσο γρηγορότερα γίνεται σαν addendums στην SQL-92. Στην Αμερική, αυτά επεξεργάζονται μόνο σαν Διεθνή Έργα (International - "I"). Η SQL/CLI ολοκληρώθηκε το 1995. Η SQL/PSM πρέπει να έχει ολοκληρωθεί στα τέλη του 1996.

Επιπρόσθετα της SQL3 εργασίας, ένας αριθμός επιπλέον έργων are persued:

- SQL/MM Μια προσπάθεια να οριστούν πρότυπα πακέτα πολυμέσων χρησιμοποιώντας τις δυνατότητες της SQL3 ADT.
- Πρόσβαση σε απομακρυσμένα δεδομένα - Remote Data Access (RDA)

Επιτροπές Προτύπων και Διαδικασιών Υπάρχει πραγματικά μεγάλος αριθμός επιτροπών προτύπων SQL σε όλο τον κόσμο. Υπάρχει ένα διεθνές γκρουπ του προτύπου SQL σαν τμήμα του ISO. Πολλές χώρες έχουν επιτροπές που ασχολούνται με την SQL. Αυτές οι χώρες (συνήθως) στέλνουν αντιπροσώπους στις συναντήσεις ISO/IEC JTC1/SC 21/WG3 DBL. Οι χώρες που συμμετέχουν ενεργά στη διαδικασία του προτύπου ISO SQL είναι οι:

- Αυστραλία
- Βραζιλία
- Καναδάς
- Γαλλία
- Γερμανία
- Ιαπωνία
- Κορέα
- Ολλανδία
- Αγγλία
- Αμερική

NIST Νομιμοποίηση

Οι υλοποιήσεις της SQL νομιμοποιούνται (στην Αμερική) από το Εθνικό Ινστιτούτο Προτύπων και Ελέγχου (NIST). Το NIST έχει για την ώρα ένα σύνολο από τεστ νομιμότητας για το entry level SQL-92. Οι ακριβείς πληροφορίες για τις απαιτήσεις της NIST νομιμοποίησης καθορίζονται ως Ομοσπονδιακό Πρότυπο Επεξεργασίας Πληροφορίας - Federal Information Processing Standard (FIPS). Οι τρέχουσες απαιτήσεις για την SQL ορίζονται στο FIPS 127-2. Μπορείτε να πάρετε τις τρέχουσες Postscript και Text εκδόσεις αυτού του κειμένου από το NIST. Η λίστα των τρεχόντων νομιμοποιημένων SQL προϊόντων μπορεί επίσης να βρεθεί στο NIST.

Εκδόσεις και Άρθρα σχετικά με το πρότυπο της SQL

Υπάρχουν δύο εκδόσεις σχετικά με το πρότυπο SQL. Και οι δύο διατίθενται από το ANSI:

- ISO/IEC 9075:1992, "Information Technology — Database Languages — SQL"
- ANSI X3.135-1992, "Database Language SQL"

Οι δυο εκδόσεις του προτύπου SQL είναι ολόιδιες εκτός από το εξώφυλλο και τις αναφορές σε άλλα πρότυπα. Και οι δύο διατίθενται από το :

```
American National Standards Institute
1430 Broadway
New York, NY 10018
USA
Phone (sales): +1.212.642.4900
```

Επιπρόσθετα στο πρότυπο SQL-92, υπάρχει ένα Technical Corrigendum (διορθώσεις bug):

```
* Technical Corrigendum 1:1994 to ISO/IEC 9075:1992
```

Το TC 1 πρέπει να διατίθεται επίσης από το ANSI. Υπάρχει μόνο η ISO έκδοση του TC 1 – εφαρμόζεται και στην ISO και ANSI έκδοση του SQL-92.

Επιπρόσθετα, έχουν γραφτεί αρκετά βιβλία σχετικά με το πρότυπο 1992 SQL. Αυτά τα βιβλία παρέχουν μια καλύτερα αναγνώσιμη περιγραφή του προτύπου από ότι το ίδιο το πρότυπο.

Σχετιζόμενα πρότυπα

Ενας σύνολο προτύπων ενδιαφέρουν την SQL κοινότητα. Αυτό το τμήμα περιέχει δείκτες σε πληροφορίες για αυτές τις προσπάθειες. Αυτοί οι δείκτες θα αυξηθούν καθώς έξτρα πληροφορίες διατίθενται στο web.

- Περιβάλλοντα SQL (FIPS 193)
- Repository Συστήματα Επόμενης Γενιάς (X3H4) - μια News Release που ζητά συμμετοχή στο "Αναπτύσσοντας Πρότυπα για Repository Συστήματα Επόμενης Γενιάς."

23.11 Τμήμα 2 - ISO/ANSI SQL Foundation

Ένα σημαντικό τμήμα της SQL3 προσπάθειας βρίσκεται στο κείμενο SQL Foundation :

- Βασικές δυνατότητες SQL/PSM (moved form SQL/PSM-92)
- Νέοι τύποι δεδομένων
- Triggers
- Υποπίνακες
- Αφηρημένοι Τύποι Δεδομένων - Abstract Data Types (ADT)
- Αντικειμενοστραφείς Δυνατότητες

Υπάρχουν αρκετά προαπαιτούμενα στις αντικειμενοστραφείς δυνατότητες:

- Ικανότητα να ορίζετε σύνθετες λειτουργίες
- Αποθήκευση σύνθετων λειτουργιών στη βάση
- Κλήση εξωτερικών διαδικασιών • Μερικές λειτουργίες μπορεί να μην είναι σε SQL, ή μπορεί να απαιτούν εξωτερικές παρεμβάσεις

Αυτές οι δυνατότητες ορίζονται σαν τμήμα του SQL/PSM

Γίνεται μια σημαντική εργασία για να εκκαθαριστεί το μοντέλο αντικειμένου SQL-3 και να ευθυγραμμιστεί με το μοντέλο αντικειμένου που προτάθηκε από το ODMG. Αυτή η προσπάθεια περιγράφεται στο X3H2 και ISO DBL paper: Accomodating SQL3 and ODMG. Είναι επίσης διαθέσιμη μια πρόσφατη έκδοση στο SQL3/OQL Merger.

SQL3 Timing

Η εργασία πάνω στο SQL3 βρίσκεται σε εξέλιξη, αλλά τα τελικά πρότυπα θα χρειαστούν χρόνια ακόμη για να ολοκληρωθούν.

- Διεθνής ψηφοφορία για την πρόοδο του SQL3 Foundation από το Πρόχειρο Έντυπο Εργασίας ως το Πρόχειρο Έντυπο Επιτροπής (Committee Draft - CD) έλαβε χώρα το φθινόπωρο του 1996.
- Η ψηφοφορία αναμένεται να προκαλέσει πολλά σχόλια

- Είναι πιθανό να ζητηθεί μια δεύτερη ψηφοφορία CD
- Μια Draft Ψηφοφορία Διεθνών Προτύπων είναι πιθανό να γίνει στα μέσα του 1998
- Το Διεθνές Πρότυπο μπορεί να ολοκληρωθεί στα μέσα του 1999.

Η ANSI έκδοση του προτύπου έχει παρόμοιο χρονοδιάγραμμα.

23.12 Τμήμα 3 - ISO/ANSI SQL Call Level Interface

Το SQL/CLI είναι ένα προγραμματιστικό interface επιπέδου κλήσεων για SQL βάσεις. Έχει σχεδιαστεί για να υποστηρίζει πρόσβαση σε βάσεις από shrink-wrapped εφαρμογές. Το CLI δημιουργήθηκε αρχικά από μια υποεπιτροπή του SQL Access Group (SAG). Οι προδιαγραφές του SAG/CLI εκδόθηκαν σαν τις προδιαγραφές του Microsoft Open DataBase Connectivity (ODBC) το 1992. Το 1993, το SAG έστειλε το CLI στις επιτροπές ANSI και ISO SQL. (Το γκρουπ SQL Access έχει συγχωνευθεί τώρα με το X/Open consortium.)

Το SQL/CLI παρέχει ένα διεθνές πρότυπο για :

- CLI ανεξάρτητο υλοποίησης, για πρόσβαση σε βάσεις SQL
- Client-Server εργαλεία που μπορούν να έχουν πρόσβαση σε μια βάση μέσω δυναμικών βιβλιοθηκών
- Υποστήριξη και ενθάρρυνση πλούσιου συνόλου εργαλείων Client-server

SQL/CLI Timing

Στα πλαίσια της διαδικασία τυποποίησης, το SQL/CLI επεξεργάζεται με μεγάλη προτεραιότητα.

- Το SQL/CLI είναι ένα συμπλήρωμα του προτύπου 1992 SQL (SQL-92)
- Ολοκληρώθηκε σαν πρότυπο ISO το 1995
- ISO/IEC 9075-3:1995 Information technology – Database languages – SQL – Part 3: Call-Level Interface (SQL/CLI)
- Η τρέχουσα SQL/CLI προσπάθεια που γίνεται είναι να προστεθεί υποστήριξη για χαρακτηριστικά της SQL3

23.13 Τμήμα 4 - ISO/ANSI SQL Persistent Stored Modules

Το SQL/PSM επεκτείνει την SQL προσθέτοντας:

- Δηλωτικές/Διαδικαστικές επεκτάσεις
- Multi-statement και Stored Procedures
- Εξωτερικές συναρτήσεις και κλήσεις διαδικασιών

Εκτός του ότι είναι ένα αξιόλογο εργαλείο ανάπτυξης εφαρμογών, το SQL/PSM παρέχει τη θεμελιώδη υποστήριξη για αντικειμενοστραφείς δυνατότητες στην SQL3.

Multi-statement και Stored Procedures

Οι Multi-statement και stored procedures προσφέρουν μια ποικιλία πλεονεκτημάτων σε ένα περιβάλλον client/server:

- Απόδοση- Μια και μια stored procedure μπορεί να εκτελέσει πολλαπλές SQL εντολές, μειώνεται η αλληλεπίδραση με τον πελάτη (client).
- Ασφάλεια - Ενας χρήστης μπορεί να αποκτήσει το δικαίωμα να καλεί μια stored procedure που ενημερώνει ένα πίνακα ή ένα σύνολο πινάκων αλλά μην έχει το δικαίωμα της απευθείας ενημέρωσης των πινάκων
- Επαναχρησιμοποιήσιμος κώδικας - Ο κώδικας σε μια stored procedure δε χρειάζεται να ξαναγραφτεί και να ξαναελεγχθεί για κάθε client εργαλείο που έχει πρόσβαση στη βάση.
- Έλεγχος - Παρέχει ένα μόνο σημείο ορισμού και ελέγχου της λογικής της εφαρμογής.

Δηλωτικές/Διαδικαστικές επεκτάσεις

Η Διαδικαστική Γλώσσα προσθέτει την ισχύ μιας παραδοσιακής γλώσσας προγραμματισμού στην SQL μέσω εντολών ελέγχου ροής και μια ποικιλία άλλων προγραμματιστικών κατασκευών.

Εντολές Ελέγχου ροής

- If-then-else
- Looping constructs
- Exception handling
- Case statement
- Begin-End blocks

Οι διαδικαστικές επεκτάσεις περιλαμβάνουν και άλλες προγραμματιστικές κατασκευές:

- Δηλώσεις μεταβλητών
- Εντολές Set για ανάθεση τιμών
- Λήψη διαγνωστικών για τις διεργασίες και πληροφορίες κατάστασης

Επιπλέον, όλες οι παραδοσιακές SQL εντολές μπορούν να συμπεριληφθούν σε διαδικασίες multi-statement.

Κλήσεις Εξωτερικών Διαδικασιών και Συναρτήσεων Ένα χαρακτηριστικό που αναφέρεται συχνά στις ευχετήριες λίστες του τι θα επιθυμούσαμε για πολλά προϊόντα βάσεων, και που έχει υλοποιηθεί σε μερικές, είναι η δυνατότητα να εμπλουτισμού των ενσωματωμένων χαρακτηριστικών με κλήσεις σε διαδικασίες που έχει γράψει ο χρήστης εξωτερικά εκτός της βάσης.

- Επιτρέπει σε ένα συγκεκριμένο site ή εφαρμογή να προσθέσει τις δικές του συναρτήσεις για τη βάση
- Μπορεί να χρησιμοποιηθεί από τις εφαρμογές βάσεις δεδομένων

Το πλεονέκτημα αυτής της δυνατότητας είναι ότι δίνει στη βάση (συνεπώς και στις εφαρμογές βάσεων) πρόσβαση σε τεράστιο αριθμό διαδικασιών και συναρτήσεων.

Χρονισμός του SQL/PSM

Το SQL/PSM έχει εν συντομία:

- Το SQL/PSM είναι μια προσθήκη στο SQL-92
- Διεθνής ψηφοφορία για την πρόοδο του SQL/PSM από το Draft Διεθνές Πρότυπο στο Διεθνές Πρότυπο ολοκληρώθηκε τον Ιανουάριο του 1996.

- Η διορθωτική συνεδρίαση το Μάιο του 1996 δεν επέλυσε όλα τα σχόλια
- Η συνέχιση της διορθωτικής συνεδρίασης PSM προγραμματίστηκε από τις 30 Σεπτεμβρίου μέχρι τις 4 Οκτωβρίου του 1996
- Το χρονοδιάγραμμα είναι πιεστικό αλλά υπάρχει μια πιθανότητα να εκδοθεί το PSM μέσα στο 1996.
- Ο επίσημος προσδιορισμός θα είναι: ISO/IEC DIS 9075-4:199? Information technology – Database languages – SQL – Part 4: SQL Persistent Stored Modules (SQL/PSM)
- Έχει ήδη ξεκινήσει δουλειά για την προσθήκη υποστήριξης SQL/PSM στα χαρακτηριστικά της SQL3.

23.14 Τμήμα 5 - Τυποποιήσεις - Δεσμεύσεις ISO/ANSI SQL/ ISO/ANSI SQL/Bindings

Για ευκολία αναφορών, οι δεσμεύσεις, τυποποιήσεις της γλώσσας προγραμματισμού έχουν συγκεντρωθεί σε ένα ξεχωριστό κείμενο. Η τρέχουσα έκδοση είναι απλά ένα απόσπασμα από τα πρότυπα της SQL-92(δυναμικά (dynamic) και ενσωματούμενα (embedded)).

Παραμένει αδιευκρίνιστη μια ποικιλία θεμάτων τυποποίησης της γλώσσας.

Στην παραδοσιακή προγραμματιστική γλώσσα, υπάρχουν οι απεικονίσεις των τύπων δεδομένων της SQL-92. Παρόλα αυτά, θα πρέπει να οριστούν και οι απεικονίσεις μεταξύ των αντικειμένων της SQL και των μεταβλητών της προγραμματιστικής γλώσσας..

Για τις αντικειμενοστραφείς γλώσσες, πρέπει να οριστεί η απεικόνιση για τους τύπους δεδομένων της SQL και μεταξύ του μοντέλου αντικειμένων της SQL και του μοντέλου αντικειμένων της αντικειμενοστραφούς γλώσσας.

Το μοντέλο δεδομένων θα πρέπει να οριστικοποιηθεί πριν αρχίσει να γίνεται αναφορά σε τέτοιου τύπου προβλήματα.

Οι τυποποιήσεις της γλώσσας θα ολοκληρωθούν ως μέρους του πρότυπου SQL3.

23.15 Τμήμα 6 - ISO/ANSI SQL XA Interface Specialization (SQL/XA)

Οι προδιαγραφές αυτές θα τυποποιήσουν ένα προγραμματιστικό interface (API) μεταξύ ενός γενικού Διαχειριστή Συνδιαλλαγών (Transaction Manager) και ενός Διαχειριστή Πόρων (Resource Manager) SQL. Θα τυποποιήσει τις κλήσεις συναρτήσεων, βασιζόμενος στα πρότυπα του ISO/IEC 10026, "Distributed Transaction Processing", που ένας Διαχειριστής Πόρων SQL θα πρέπει να υποστηρίζει, για να κάνει commit δύο σταδίων (two-phase commit) Το βασικό κείμενο αναφοράς εξάγεται από μια δημοσίευση του X/Open, με την άδεια του X/Open, που προσδιορίζει ρητά τις παραμέτρους εισόδου και εξόδου και την σημειογραφία (semantics), για τις ακόλουθες συναρτήσεις: xa_close, xa_commit, xa_complete, xa_end, xa_forget,xa_open, xa_prepare, xa_recover, xa_rollback, και xa_start.

Ο ISO επιχειρεί να επιταχύνει τις προδιαγραφές του X/Open XA. Η διαδικασία επιτάχυνσης υιοθετεί βιομηχανικές προδιαγραφές χωρίς αλλαγές. Η ψηφοφορία για το XA στο ISO SC21, JTC επίπεδο 1, άρχισε στις 27 Απριλίου, 1995 και τελείωσε στις 27 Οκτωβρίου, 1995. Αν το 75 των ψηφοφόρων εγκρίνουν τις προδιαγραφές του XA, και από τα 2/3 των p-μελών του JTC 1, θα αποτελέσει ένα Διεθνές Πρότυπο. Αν η ψηφοφορία το εγκρίνει, το SQL/XA θα μπορούσε να γίνει ένα πρότυπο μέσα στο 1996.

23.16 Part 7 - ISO/ANSI SQL Temporal

Η Temporal SQL ασχολείται με χρονικά εξαρτώμενα δεδομένα. Η ιδέα είναι πως είναι χρήσιμο να αναλύουμε τα δεδομένα για να δούμε πως έμοιαζαν κάποια δεδομένη χρονική στιγμή. Η Temporal SQL είναι ένα paper του Δεκεμβρίου του 1994 από τον Rick Snodgrass που περιγράφει αυτές τις έννοιες.

Η X3 ανακοινώνει την έγκριση ενός Νέου Έργου, ISO/IEC 9075 Μέρος 7ο: Η SQL/Temporal είναι μια δημοσίευση τύπου σχετική με την SQL/Temporal.

```
-------------------------------------------------------------------------
                              Temporal SQL
                              ************
Rick Snodgrass (\textgreek{pr'oedroc} \textgreek{thc} \textgreek{epitrop'hc} TSQL2)
31-\textgreek{Dek}-1994
```

Αρκετοί αμφισβήτησαν την ανάγκη αυξημένης υποστήριξης χρόνου στην SQL3 (όπως αυτό προτείνεται από το DBL RIO-75, όπου και ζητά ένα νέο τμήμα της SQL που να υποστηρίζει χρονικές (temporal) βάσεις). Ο ισχυρισμός τους βασίζεται στο ότι οι αφηρημένοι τύποι δεδομένων (abstract data types -ADT's) αρκούν για χρονική υποστήριξη. Σε αυτό το πληροφοριακό κείμενο, διαφωνώ, και, χρησιμοποιώντας αδιάσειστα παραδείγματα, δείχνω πως χρησιμοποιώντας μόνο στήλες με αφηρημένους τύπους δεδομένων δεν αρκεί για χρονικές ερωτήσεις. Ειδικότερα, πολλές κοινές χρονικές ερωτήσεις, είναι είτε δύσκολες να προσομοιωθούν με SQL, είτε απαιτούν την ενσωμάτωση SQL σε μια διαδικαστική (procedural) γλώσσα. Οι εναλλακτικές λύσεις εκφράζονται σε TSQL2, μια χρονική επέκταση στην SQL-92.

23.16.1 ΕΙΣΑΓΩΓΗ

Η υποστήριξη έγκυρου χρόνου (Valid-time) ξεπερνάει αυτή ενός κοινού χρονικού αφηρημένου τύπου δεδομένων (ADT). Στον τελευταίο, μια στήλη προσδιορίζεται σαν ένα χρονικό πεδίο, είτε ως DATE είτε ως INTERVAL (θα δοθούν παραδείγματα σύντομα). Με τον έγκυρο χρόνο, οι γραμμές ενός πίνακα διαφοροποιούνται με το πέρασμα του χρόνου, καθώς διαφοροποιείται και η πραγματικότητα. Η χρονική σφραγίδα (timestamp) που σχετίζεται με μια γραμμή ενός πίνακα έγκυρων χρόνων (valid-time table) μεταφράζεται από την γλώσσα ερωτήσεων ως η χρονική στιγμή για την οποία ο συνδυασμός τιμών των στηλών της γραμμής εκείνης ήταν έγκυρος. Η ύπαρξη της έμμεσης αυτής χρονικής σφραγίδας επιτρέπει την διατύπωση σαφών και πρωτότυπων ερωτήσεων.

23.16.2 ΜΙΑ ΜΕΛΕΤΗ- ΑΠΟΘΗΚΕΥΣΗ ΣΥΓΧΡΟΝΩΝ/ΤΡΕΧΟΥΣΩΝ ΠΛΗΡΟΦΟΡΙΩΝ

Το Γραφείο Προσωπικού του Πανεπιστημίου της Αριζόνα έχει κάποιες πληροφορίες σε μια βάση δεδομένων, μέσα στις οποίες και το όνομα των υπαλλήλων, τον τρέχοντα μισθό τους, και τον τρέχουσα θέση τους. Αυτό μπορεί να περιγραφεί με έναν απλό πίνακα.

```
Employee(Name, Salary, Title)
```

Δεδομένου αυτού του πίνακα, η εύρεση του μισθού ενός υπαλλήλου είναι εύκολη.

```
SELECT Salary
FROM Employee
WHERE Name = 'Bob'
```

Τώρα το Γραφείο Προσωπικού επιθυμεί να καταγράφει και την ημερομηνία γέννησης. Για να το κάνει, προσθέτει μια στήλη στον πίνακα, και έτσι προκύπτει το ακόλουθο σχήμα.

```
Employee(Name, Salary, Title, DateofBirth DATE)
```

Η εύρεση των γενεθλίων ενός υπαλλήλου είναι ανάλογη με το να βρούμε το μισθό του.

```
SELECT DateofBirth
FROM Employee
WHERE Name = 'Bob'
```

23.16.3 ΜΕΛΕΤΗ - ΑΠΟΘΗΚΕΥΣΗ ΙΣΤΟΡΙΚΩΝ ΠΛΗΡΟΦΟΡΙΩΝ

Το Γραφείο Προσωπικού επιθυμεί περάσει τα ιστορικά στοιχεία των υπαλλήλων σε ηλεκτρονική μορφή. Για να το κάνει, προσθέτουν δύο στήλες, μια που υποδεικνύει πότε έγινε έγκυρη η πληροφορία της γραμμής, και μια για το πότε έπαψε να ισχύει.

Employee (Name, Salary, Title, DateofBirth, Start DATE, Stop DATE)

Στο μοντέλο δεδομένων, αυτές οι δύο στήλες είναι παρόμοιες με το DateofBirth. Παρόλα αυτά, η παρουσία τους έχει επιπτώσεις μεγάλης κλίμακας.

23.16.4 ΜΕΛΕΤΗ - ΠΡΟΒΟΛΗ (PROJECTION)

Για να βρούμε τον τρέχοντα μισθό ενός υπαλλήλου, τα πράγματα είναι πλέον δυσκολότερα.

```
SELECT Salary
FROM Employee
WHERE Name = 'Bob' \textgreek{kai} Start <= CURRENT_DATE \textgreek{kai} CURRENT_DATE <= Stop
```

Αυτή η ερώτηση είναι πιο σύνθετη από ότι η προηγούμενη. Η δυσκολία είναι προφανώς οι δύο νέες στήλες. Το Γραφείο Προσωπικού θέλει να διανέμει σε κάθε υπάλληλο το μισθολογικό του αρχείο. Ειδικότερα, για κάθε έναν, θα πρέπει να καθοριστούν τα μέγιστα διαστήματα ανά μισθό. Δυστυχώς, αυτό δεν είναι δυνατό με SQL. Ένας υπάλληλος έχει ένα τυχαίο αριθμό αλλαγών θέσεων μεταξύ μισθολογικών αλλαγών.

```
Name    Salary  Title             DateofBirth  Start       Stop
----    ------  -----             -----------  -----       ----
Bob     60000   Assistant Provost 1945-04-09   1993-01-01  1993-05-30
Bob     70000   Assistant Provost 1945-04-09   1993-06-01  1993-09-30
Bob     70000   Provost           1945-04-09   1993-10-01  1994-01-31
Bob     70000   Professor         1945-04-09   1994-02-01  1994-12-31
```

\textgreek{Eik'ona} 1

Σημειώστε πως υπάρχουν τρεις γραμμές όπου ο μισθός του Bob παρέμεινε σταθερός στις \$70,000. Γι αυτό το αποτέλεσμα θα έπρεπε να δώσει δύο γραμμές για τον Bob.

```
Name    Salary  Start         Stop
----    ------  -----         ----
Bob     60000   1993-01-01    1993-05-30
Bob     70000   1993-06-01    1994-12-31
```

Μια εναλλακτική λύση είναι να δοθεί στον χρήστη μια εκτύπωση πληροφοριών Μισθού και Τίτλου, και να αφήνουμε τον χρήστη να καθορίζει πότε άλλαξε ο μισθός του. Η εναλλακτική αυτή δεν είναι πολύ θελκτική ή ρεαλιστική. Μια δεύτερη, είναι να χρησιμοποιήσουμε όσο πιο πολύ SQL γίνεται.

```
CREATE TABLE Temp(Salary, Start, Stop)
AS      SELECT Salary, Start, Stop
        FROM Employee;
```

επαναλάβατε

```
UPDATE Temp T1
SET (T1.Stop) = (SELECT MAX(T2.Stop)
                 FROM Temp AS T2
```

```
                    WHERE T1.Salary = T2.Salary \textgreek{kai} T1.Start < T2.Start
                          \textgreek{kai} T1.Stop >= T2.Start \textgreek{kai} T1.Stop < T2.Stop)
        WHERE EXISTS (SELECT *
                     FROM Temp AS T2
                     WHERE T1.Salary = T2.Salary \textgreek{kai} T1.Start < T2.Start
                          \textgreek{kai} T1.Stop >= T2.Start \textgreek{kai} T1.Stop < T2.Stop)
        \textgreek{m'eqri} \textgreek{na} \textgreek{mhn} \textgreek{enhmerwje'i} \textgreek{kami'a} \text

DELETE FROM Temp T1

WHERE EXISTS (SELECT *
             FROM Temp AS T2
             WHERE T1.Salary = T2.Salary
                  \textgreek{kai} ((T1.Start > T2.Start \textgreek{kai} T1.Stop <= T2.Stop)
                  OR (T1.Start >= T2.Start \textgreek{kai} T1.Stop < T2.Stop))
```

Ο βρόγχος βρίσκει τα διαστήματα εκείνα που επικαλύπτονται ή είναι γειτονικά και γι αυτό θα έπρεπε να συγχωνευθούν. Ο βρόχος εκτελείται N φορές στην χειρότερη περίπτωση, όπου το N είναι ο αριθμός των γραμμών σε μια αλυσίδα επικαλυπτόμενων ή γειτονικών γραμμών με ίσες τιμές. Ο αναγνώστης μπορεί να προσομοιώσει την ερώτηση του πίνακα του παραδείγματος για να πειστεί για την ορθότητα του παραπάνω.

Μια τρίτη εναλλακτική είναι να χρησιμοποιήσουμε SQL μόνο για να ανοίξουμε ένα κέρσορα (cursor) στον πίνακα. Μια συνδεδεμένη λίστα χρονικών περιόδων θα κρατηθεί, κάθε μια έχοντας και ένα μισθό. Η συνδεδεμένη λίστα αρχικοποιείται κενή.

```
DECLARE emp_cursor CURSOR FOR
        SELECT Salary, Title, Start, Stop
        FROM Employee;
OPEN emp_cursor;
loop:
        FETCH emp_cursor INTO :salary, :start, :stop;
        if no-data returned then goto finished;
        find position in linked list to insert this information;
        goto loop;
finished:
CLOSE emp_cursor;
```

διέτρεξε τη λίστα τυπώνοντας ημερομηνίες και μισθούς

Η συνδεδεμένη λίστα μπορεί και να μην χρειάζεται στην περίπτωση αυτή ανά ο κέρσορας είναι ORDER BY Start.

Σε κάθε περίπτωση, η ερώτηση, που είναι και φυσική, είναι δύσκολο να εκφραστεί με τις υπάρχουσες δυνατότητες της SQL-92. Η ερώτηση είναι τετριμμένη σε TSQL2.

```
SELECT Salary
FROM Employee
```

23.16.5 ΜΕΛΕΤΗ - ΣΥΣΧΕΤΙΣΜΟΣ (JOIN)

Μια πιο δραστική προσέγγιση είναι να αποφύγουμε το πρόβλημα της εξαγωγής της μισθολογικής ιστορίας με αναδιοργάνωση του σχήματος ώστε να αποσυζεύξουμε το μισθό, τη θέση και την ημερομηνία γέννησης (παρακάτω, αγνοούμε την ημερομηνία γέννησης, για απλότητα).

```
Employee1 (Name, Salary, Start DATE, Stop DATE)
Employee2 (Name, Title, Start DATE, Stop DATE)
```

Ο πίνακας Employee1 έχει ως ακολούθως.

```
Name    Salary  Start           Stop
----    ------  -----           ----
Bob     60000   1993-01-01      1993-05-30
Bob     70000   1993-06-01      1993-12-31
```

Εδώ είναι και ο πίνακας Employee2 του παραδείγματος.

```
Name    Title                   Start           Stop
----    ------                  -----           ----
Bob     Assistant Provost       1993-01-01      1993-09-30
Bob     Provost                 1993-10-01      1994-01-31
Bob     Professor               1994-02-01      1994-12-31
```

Με αυτή την αλλαγή, η απόκτηση της πληροφορίας του μισθού για έναν υπάλληλο είναι πλέον εύκολη.

```
SELECT Salary, Start, Stop
FROM Employee1
WHERE Name = 'Bob'
```

Αλλά τι θα γίνει εάν το Γραφείο Προσωπικού θελήσει έναν πίνακα μισθού, διαστημάτων θέσης (δηλαδή, ας υποθέσουμε πως το Γ.Π. θέλει έναν πίνακα της μορφής της Εικόνας 1); Μια εναλλακτική λύση είναι να τυπώσουμε τους δύο πίνακες και να αφήσουμε τον χρήστη να υπολογίσει τους συνδυασμούς. Μια δεύτερη εναλλακτική, είναι να χρησιμοποιήσουμε μόνο SQL. Δυστυχώς, η ερώτηση αυτή πρέπει να κάνει μια ανάλυση καταστάσεων του πως μια γραμμή του Employee1 επικαλύπτει γραμμές του Employee2. Υπάρχουν τέσσερις δυνατές καταστάσεις.

```
SELECT Employee1.Name, Salary, Dept, Employee1.Start, Employee1.Stop
FROM Employee1, Employee2
WHERE Employee1.Name = Employee2.Name
     and Employee2.Start <= Employee1.Start and Employee1.Stop < Employee2.Stop
UNION
SELECT Employee1.Name, Salary, Dept, Employee1.Start, Employee2.Stop
FROM Employee1, Employee2
WHERE Employee1.Name = Employee2.Name
     and Employee1.Start >= Employee2.Start and Employee2.Stop < Employee1.Stop
        \textgreek{kai} Employee1.Start < Employee2.Stop
UNION
SELECT Employee1.Name, Salary, Dept, Employee2.Start, Employee1.Stop
FROM Employee1, Employee2
WHERE Employee1.Name = Employee2.Name
     and Employee2.Start > Employee1.Start and Employee1.Stop < Employee2.Stop
        and Employee2.Start < Employee1.Stop
UNION
SELECT Employee1.Name, Salary, Dept, Employee2.Start, Employee2.Stop
FROM Employee1, Employee2
WHERE Employee1.Name = Employee2.Name
     and Employee2.Start > Employee1.Start and Employee2.Stop < Employee1.Stop
```

Το να πιάσουμε τις τέσσερις καταστάσεις είναι μια δύσκολη και ενδιαφέρουσα εργασία. Σε TSQL2, το να κάνουμε ένα χρονικό συσχετισμό (temporal join) είναι όπως κάποιος θα περίμενε.

```
SELECT Employee1.Name, Salary, Dept
FROM Employee1, Employee2
WHERE Employee1.Name = Employee2.Name
```

23.16.6 ΜΕΛΕΤΗ - ΑΘΡΟΙΣΤΙΚΟΙ ΤΕΛΕΣΤΕΣ (AGGREGATES)

Τώρα το Γ.Π. ερωτάται, ποιος είναι ο μέγιστος μισθός; Πριν την εισαγωγή του χρόνου, αυτό ήταν εύκολο.

```
SELECT MAX(Salary)
FROM Employee
```

Τώρα όμως που η μισθολογική ιστορία αποθηκεύεται, θα θέλαμε μια ιστορία του μέγιστου μισθού ανά τον χρόνο. Το πρόβλημα φυσικά είναι, πως η SQL δεν παρέχει χρονικούς αθροιστικούς τελεστές. Ο εύκολος τρόπος να γίνει το παραπάνω, είναι να τυπώσουμε την πληροφορία και να την διατρέξουμε χειροκίνητα για τα μέγιστα. Εναλλακτικά, θα μπορούσαμε να κάνουμε τεχνάσματα και να μετατρέψουμε την αθροιστική ερώτηση σε μια μια μη αθροιστική, και μετά αυτή σε χρονική. Η μη αθροιστική θα έβρισκε εκείνους τους μισθούς για τους οποίους δεν υπάρχει μεγαλύτερος μισθός.

```
SELECT Salary
FROM Employee AS E1
WHERE NOT EXISTS (SELECT *
                  FROM Employee AS E2
                  WHERE E2.Salary > E1.Salary)
```

Η μετατροπή μιας τέτοιας ερώτησης απέχει πολύ από το να θεωρηθεί τετριμμένη. Το ακόλουθο είναι μια προσέγγιση.

```
CREATE TABLE Temp (Salary, Start, Stop)
AS      SELECT Salary, Start, Stop
        FROM Employee;
INSERT INTO Temp
        SELECT T.Salary, T.Start, E.Start
        FROM Temp AS T, Employee AS E
        WHERE E.Start >= T.Start and E.Start < T.Stop and E.Salary > T.Salary;

INSERT INTO Temp
        SELECT T.Salary, T.Stop, E.Stop
        FROM Temp AS T, Employee AS E
        WHERE E.Stop > T.Start and E.Stop <= T.Stop and E.Salary > T.Salary;
DELETE FROM Temp T
WHERE EXISTS (SELECT *
              FROM Employee AS E
              WHERE ((T.Start => E.Start and T.Start < E.Stop)
                       OR (E.Start >= T.Start and E.Start < T.Stop))
                    \textgreek{kai} E.Salary > T.Salary;
```

Αυτή η προσέγγιση δημιουργεί έναν βοηθητικό πίνακα. Προσθέτουμε σε αυτόν τον πίνακα την χαμηλότερη περίοδο από μια αφαίρεση διαστημάτων και την μεγαλύτερη περίοδο από μια αφαίρεση διαστημάτων. Μετά διαγράφουμε όλες τις περιόδους που επικαλύπτονται με κάποια γραμμή που ορίζεται από την υπό-ερώτηση (subquery), και μένουν οι NOT EXISTS. Τελικά, δημιουργούμε από τον βοηθητικό πίνακα τα μέγιστα διαστήματα, με τον ίδιο τρόπο που υπολογίστηκε παραπάνω η πληροφορία μισθών. Όπως μπορούμε να φανταστούμε, τέτοιος κώδικας SQL είναι υπερβολικά αναποτελεσματικός για να εκτελεστεί, δεδομένου των σύνθετων φωλιασμένων (nested) υποερωτήσεων με τους τελεστές σύγκρισης με κριτήρια ανισότητας.

Μια τρίτη εναλλακτική λύση, είναι να χρησιμοποιήσουμε SQL όσο λιγότερο γίνεται, και να υπολογίσουμε την επιθυμητή ιστορικότητα από μέγιστες τιμές σε μια άλλη γλώσσα χρησιμοποιώντας ενσωματωμένους κέρσορες.

Η ερώτηση σε TSQL2 είναι απλούστατη.

```
SELECT MAX(Salary)
FROM Employee
```

23.16.7 ΠΕΡΙΛΗΨΗ

Οι περισσότερες εφαρμογές με βάση δεδομένων έχουν να αντιμετωπίσουν και δεδομένα με χρονική διακύμανση είναι ανάγκη να αντιμετωπιστούν . Η υποστήριξη έγκυρου-χρόνου είναι απούσα στην SQL. Πολλές κοινές χρονικές ερωτήσεις είναι λοιπόν είτε δύσκολες να εξομοιωθούν μέσω SQL, ή απαιτούν ενσωμάτωση της SQL σε μια δηλωτική / διαδικαστική γλώσσα (procedural), λόγω της έλλειψης υποστήριξης πινάκων έγκυρου-χρόνου στο μοντέλο δεδομένων και στην κατασκευή ερωτήσεων της SQL.

Σε άλλο σημείο, δείξαμε πως η υποστήριξη έγκυρου-χρόνου απαιτεί μικρές αλλαγές στην υλοποίηση της DBMS, και μπορεί να απλοποιήσει δραματικά κάποιες ερωτήσεις, να κάνει δυνατές άλλες, και ακολούθως να επιτρέψει μεθόδους δεικτοδότησης και στρατηγικές βελτιστοποιήσεων που μπορούν να οδηγήσουν σε σημαντικές καλυτερεύσεις απόδοσης.

Με το νέο τμήμα της SQL3 που υποστηρίζει πληροφορίες με χρονική διακύμανση, μπορούμε να αρχίσουμε να στοχεύουμε σε τέτοιες εφαρμογές, επιτρέποντας στην SQL3 να διαχειρίζεται καλύτερα τα χρονικά δεδομένα.

```
----------------------------------------------------------------------------
         \textgreek{Epitrop'h} \textgreek{Epik'urwshc} \textgreek{Prot'upwn}* X3, \textgreek{Teqnolog'ia
\textgreek{DHMOSIEUSH} \textgreek{NEWN}

\textgreek{Ar}. \textgreek{Eggr'afou}:       PR/96-0002

\textgreek{Ap'anthsh} \textgreek{eic}:       Barbara Bennett \textgreek{sto} bbennett@itic.nw.dc.us

          \textgreek{H} X3 \textgreek{anakoin'wnei} \textgreek{thn} \textgreek{'Egkrish} \textgreek{en'o

                      9075 \textgreek{Tm'hma} 7:  SQL/Temporal

Washington D.C., \textgreek{Ianou'arioc} 1996
----------------------------------------------------------------------------
```

– Επιτροπή Επικύρωσης Προτύπων X3, Τεχνολογία Πληροφοριών ανακοινώνει την έγκριση ενός Νέου Έργου στην χρονική υποστήριξη SQL (SQL/Temporal), ISO/IEC 9075 Τμήμα 7. Ο σκοπός του προτεινόμενου προτύπου προσδιορίζει ένα νέο Τμήμα στο ανερχόμενο πρότυπο SQL3, π.χ., το Τμήμα 7, Χρονική SQL, ως επεκτάσεις της γλώσσας SQL που υποστηρίζουν την ανάκτηση , και επεξεργασία χρονικών δεδομένων σε ένα περιβάλλον μιας SQL βάσης δεδομένων. Η επόμενη συνεδρία X3H2 προγραμματίστηκε για τις 11-14 Μαρτίου του 1996 στο Κάνσας.

Ερωτήσεις για το έργο αυτό θα πρέπει να σταλούν στο

```
Chairman of X3H2,
Dr. Donald R. Deutsch,
Sybase, Inc., Suite 800,
6550 Rock Spring
Drive, Bethesda, MD  20817.
Email: deutsch@sybase.com.
```

Μια αρχική πρόσκληση για πατέντες και σχετικά θέματα (πνευματικές ιδιοκτησίες, εμπορικά σήματα) (copyrights, trademarks) εκδίδεται αυτόν τον καιρό. Υποβάλετε πληροφορίες για τα θέματα αυτά στο

```
X3 Secretariat at
```

```
1250 Eye Street
NW, Suite 200,
Washington DC  20005.
Email: x3sec@itic.nw.dc.us
FAX:  (202)638-4922.
```

23.17 Τμήμα 8 - ISO/ANSI ΠΟΛΥΜΕΣΑ SQL (SQL MULTIMEDIA - SQL/MM)

Ένα νέο διεθνές έργο τυποποίησης ISO/IEC για την ανάπτυξη μιας βιβλιοθήκης κλάσεων SQL (SQL class library) για εφαρμογές πολυμέσων εγκρίθηκε από τις αρχές του 1993. Αυτή η νέα δραστηριότητα τυποποίησης, ονομαζόμενη SQL Multimedia (SQL/MM), θα καθορίσει πακέτα με ορισμούς αφηρημένων τύπων δεδομένων SQL (abstract data type- ADT) χρησιμοποιώντας τις ευκολίες για τον καθορισμό του ADT και την πρόκληση που παρέχεται στις αναμενόμενες προδιαγραφές του SQL3. Το SQL/MM έχει σαν στόχο να τυποποιήσει τις βιβλιοθήκες κλάσεων για την επιστήμη και μηχανική, την full-text επεξεργασία κειμένου, και τις μεθόδους για διαχείριση αντικειμένων πολυμέσων όπως εικόνα, ήχος, animation, μουσική, και βίντεο. Θα παρέχει πιθανά ένα SQL language binding αντικείμενο πολυμέσων από άλλες JTC1 οργανώσεις προτυποποίησης (π.χ. SC18 για κείμενα, SC24 για εικόνες, και SC29 για φωτογραφίες και κινούμενες εικόνες).

Το πλάνο αυτού του έργου για το SQL/MM δείχνει ότι θα είναι ένα σπονδυλωτό πρότυπο που θα αποτελείται από ένα αυξανόμενο αριθμό τμημάτων. Το τμήμα 1 θα είναι ένα Πλαίσιο (Framework) που θα προσδιορίζει πως θα κατασκευαστούν τα υπόλοιπα μέρη. Κάθε ένα από τα υπόλοιπα μέρη, θα είναι αφοσιωμένο σε ένα συγκεκριμένο πακέτο SQL εφαρμογών. Η παρακάτω δομή τμημάτων του SQL/MM υπάρχει από τον Αύγουστο του 1994:

- Τμήμα 1: Πλαίσιο. Μη τεχνική περιγραφή του πως δομείται το κείμενο.
- Τμήμα 2: Μέθοδοι Κειμένου και ADTs για επεξεργασία δεδομένων κειμένου. Περί τις 45 σελίδες.
- Τμήμα 3: Χωρικές Μέθοδοι και ADTs για διαχείριση χωρικών δεδομένων. Περί τις 200 σελίδες με συνεισφορές από ειδικούς στα χωρικά δεδομένα, από οργανώσεις 3 χωρών.
- Τμήμα 4: Μεθόδους Γενικού Σκοπού και ADTs για μιγαδικούς αριθμούς. Οι δυνατότητες περιλαμβάνουν τριγωνομετρικές και εκθετικές συναρτήσεις, διανύσματα, σύνολα, κλπ. Προς το παρόν περί τις 90 σελίδες.

υπάρχει ένα πλήθος στάνταρ εργασιών στον τομέα τον Χωρικών και Γεωγραφικών Συστημάτων:

- ANSI X3L1 - Geographic Information Systems. Mark Ashworth of Unisys is the liason between X3L1 και ANSI X3H2. He is also the editor for parts 1, 3, και 4 of the SQL/MM draft.
- ISO TC 211 - Geographic information/Geomatics

24 Τεχνική υποστήριξη για την PostgreSQL

Αν έχετε οποιαδήποτε τεχνική ερώτηση ή συναντήσετε οποιοδήποτε πρόβλημα μπορείτε να στείλετε e-mail στο:

- `pgsql-questions@postgresql.org`

και να περιμένετε απάντηση με e-mail σε λιγότερο από μια μέρα. Καθώς η βάση των χρηστών στο internet είναι τεράστια, και οι χρήστες υποστηρίζουν τους άλλους χρήστες, το internet θα είναι ικανό να δώσει εύκολα τεχνική υποστήριξη σε δισεκατομμύρια χρήστες. Η υποστήριξη με Email είναι πολύ πιο βολική από την τηλεφωνική υποστήριξη καθώς μπορείτε να κάνετε cut και paste τα μηνύματα λαθών, την έξοδο του προγράμματος κλπ. και να το μεταδώσετε εύκολα στη λίστα αλληλογραφίας/newsgroup.

Στο προσεχές μέλλον, ο οργανισμός της PostgreSQL θα πουλά τεχνική υποστήριξη σε μεγάλες/μικρές εταιρείες, τα έσοδα που θα προκύψουν θα χρησιμοποιηθούν για τη συντήρηση αρκετών mirror sites (web και ftp) σε όλο τον κόσμο. Θα χρησιμοποιηθούν επίσης για την παραγωγή εκτυπωμένης τεκμηρίωσης, βοηθημάτων, βιβλίων που θα βοηθήσουν τους πελάτες.

Μπορείτε επίσης να βοηθηθείτε από συμβουλευτικές εταιρείες όπως η Anderson, WGS (Work Group Solutions). Επικοινωνήστε μαζί τους για βοήθεια -

- Work Group Solutions `<http://www.wgs.com>`
- Anderson Consulting `<http://www.ac.com>`

25 Οικονομικά και Επαγγελματικά Θέματα

Οι εμπορικές βάσεις πληρώνουν πολλούς φόρους, ενδεικτικά αναφέρουμε : ομοσπονδιακούς, πολιτειακούς, πωλήσεων, εργασιακούς, κοινωνικής ασφάλειας, ιατρικής περίθαλψης εργαζομένων, θέλγητρων για τους εργαζόμενους, κόστη για marketing και διαφήμιση. Όλα αυτά τα έξοδα δεν πηγαίνουν κατευθείαν για την ανάπτυξη της βάσης. Όταν αγοράζετε μια εμπορική βάση, μέρος της τιμής πηγαίνει για επιβαρύνσεις όπως φόροι, έξοδα και το ισοζύγιο του κόστους της βάσης R&D.

Οι εμπορικές βάσεις, επίσης, πρέπει να πληρώσουν για κτίρια/κτίσματα και να αγοράσουν μηχανές Unix, να τις εγκαταστήσουν και να τις συντηρήσουν. Όλα αυτά τα έξοδα επιβαρύνουν τελικά τον πελάτη.

Η PostgreSQL υπερέχει έναντι των εμπορικών βάσεων μια και δεν πληρώνουν άμεσα φόρους καθώς αναπτύσσεται μέσω internet. Ένα τεράστιο πλήθος ανθρώπων συνεισφέρει στην ανάπτυξη της PostgreSQL. Για παράδειγμα, σε μια υποθετική περίπτωση, αν υπάρχουν ένα εκατομμύριο εταιρείες στην Αμερική και κάθε μια συνεισφέρει 10 $ (αξία του λογισμικού PostgreSQL) τότε κάθε μία θα λάβει δέκα εκατομμύρια δολάρια!! Αυτή είναι η **ΜΑΓΕΙΑ** της ανάπτυξης λογισμικού στο internet.

Μέχρι τώρα, ο πηγαίος κώδικας της PostgreSQL είναι περίπου 2,00,000 γραμμές σε κώδικα "C", "C++". Αν το κόστος κάθε γραμμής κώδικα "C" είναι $ 2 τότε το συνολικό κόστος της PostgreSQL θα ήταν σήμερα $ 4,00,000 (τετρακόσιες χιλιάδες δολάρια!).

Πολλές εταιρείες ήδη αναπτύσσουν τεράστιες ποσότητες κώδικα "C", "C++". Έτσι, το να πάρουν τον πηγαίο κώδικα της PostgreSQL και να συνεργαστούν με άλλες εταιρείες στο internet θα ωφελήσει την εταιρεία να εξοικονομήσει χρόνο και προσπάθεια.

26 Λίστα άλλων βάσεων δεδομένων

Παρακάτω ακολουθεί λίστα άλλων SQL βάσεων δεδομένων για Unix, Linux.

- Πατήστε και πηγαίνετε στο Applications->databases. `<http://www.caldera.com/tech-ref/linuxapps/linapps.html>`
- Πατήστε και πηγαίνετε στο Applications->databases. `<http://www.xnet.com/~blatura/linapps.shtml>`
- Πηγές για Βάσεις Δεδομένων `<http://linas.org/linux/db.html>` γράφτηκε από τον Linas Vepstas: `linas@fc.net`

- Λίστα δωρεάν βάσεων <http://cuiwww.unige.ch:80/~scg/FreeDB/FreeDB.list.html>
- Λίστα RDBMS του Browne <http://www.hex.net/~cbbrowne/rdbms.html> γράφτηκε από τον Christopher B. Browne cbbrowne@hex.net
- Λίστα Σχεσιακών DBMS του SAL <http://SAL.KachinaTech.COM/H/1/>
- Λίστα Αντικειμενοστραφών DBMS του SAL <http://SAL.KachinaTech.COM/H/2/>
- Λίστα βοηθητικών εφαρμογών και άλλων βάσεως του SAL <http://SAL.KachinaTech.COM/H/3/>
- ACM SIGMOD. Δείκτης του δημόσια διαθέσιμου λογισμικού Βάσεων Δεδομένων <http://bunny.cs.uiuc.edu/sigmod/databaseSoftware/>

27 Χρήσιμες συμβουλές για αναζητήσεις στο Internet World Wide Web

Το Internet είναι απέραντο, υπάρχει σε αυτό τεράστια ποσότητα λογισμικού και μια θάλασσα πληροφοριών από κάτω του. Αυξάνεται με ρυθμό 300% ετησίως σε όλο τον κόσμο. Υπολογίζεται ότι υπάρχουν περίπου 10 εκατομμύρια Web sites σε όλη τη γη!

Για να ψάξετε για πληροφορίες θα χρησιμοποιήσετε μηχανές αναζήτησης όπως το "Yahoo", "Netscape", "Lycos" κλπ. Πηγαίνετε στο Yahoo και πατήστε στο search. Χρησιμοποιήστε τις επιλογές φιλτραρίσματος για να συγκεκριμενοποιήσετε τα κριτήρια αναζήτησης. Η default επιλογή αναζήτησης είναι η "Έξυπνη αναζήτηση" ("Intelligent search") που είναι πιο γενική και εμφανίζει όλες τις πιθανότητες. Πατήστε στις "Επιλογές" ("Options") για να διαλέξετε αναζήτηση με "ακριβή φράση"("EXACT phrase"), αναζήτηση με "AND", με "OR", κλπ. Με αυτόν τον τρόπο θα βρείτε τις πληροφορίες που θέλετε πολύ γρηγορότερα.

28 Συμπέρασμα

Μετά από έρευνα σε όλες τις διαθέσιμες βάσεις που είναι **ΔΩΡΕΑΝ** και ο πηγαίος τους κώδικας είναι διαθέσιμος, καταλήξαμε ότι ΜΟΝΟ η PostgreSQL είναι η ΠΕΡΙΣΣΟΤΕΡΟ ώριμη, ΠΕΡΙΣΣΟΤΕΡΟ ευρέως διαδεδομένη και ισχυρή, δωρεάν βάση RDBMS SQL (αντικειμενοστραφής-σχεσιακή), στον κόσμο.

Η PostgreSQL είναι ιδιαίτερα ελκυστική μια και έχει ήδη γίνει αρκετή δουλειά, και έχει οδηγούς ODBC, JDBC με τους οποίους είναι δυνατό να γράψετε εφαρμογές ανεξαρτήτως βάσης. Οι εφαρμογές που γράφονται στην PostgreSQL με χρήση οδηγών ODBC, JDBC είναι εύκολα μεταφέρσιμες σε άλλες βάσεις, όπως Oracle, Sybase και Informix. Επίσης, εφαρμογές που έχουν γραφτεί σε Oracle, Sybase και Informix με χρήση οδηγών ODBC, JDBC είναι εύκολα μεταφέρσιμες στην PostgreSQL.

Μπορεί να αναρωτηθείτε "Αλλά γιατί PostgreSQL ;" Η απάντηση είναι ότι μια και χρειάζεται πολύ περισσότερος χρόνος για να αναπτυχθεί ένα νέο σύστημα βάσης από την αρχή, είναι λογικό να διαλέξουμε ένα σύστημα βάσης δεδομένων που ικανοποιεί τις παρακάτω συνθήκες -

Ένα σύστημα βάσης δεδομένων

- του οποίου ο πηγαίος κώδικας είναι διαθέσιμος
- που δεν έχει περιορισμούς σε θέματα άδειας, ή κάποια προσαρτημένη ιδιοκτησία σε αυτό
- το οποίο διανέμεται μέσω internet
- το οποίο αναπτύσσεται εδώ και πολλά χρόνια.

- το οποίο ικανοποιεί μελλοντικές ανάγκες όπως την SQL 3 (SQL 98)
- το οποίο έχει εξελιγμένες δυνατότητες

και απλά συμβαίνει να είναι η 'PostgreSQL' που ικανοποιεί όλες αυτές τις συνθήκες και να είναι το κατάλληλο λογισμικό για αυτή την περίπτωση.

29 Συχνά Διατυπωμένες Ερωτήσεις σχετικά με την PostgreSQL

Πηγαίνετε στην τελευταία έκδοση του FAQ για General, Linux και Irix στο

- <http://www.postgresql.org/docs/faq-english.shtml>

30 Σημείωση Πνευματικών Δικαιωμάτων

A Παράρτημα A - Σύνταξη του ANSI/ISO SQL 1992

```
\textgreek{Aut'o} \textgreek{to} \textgreek{arqe'io} \textgreek{peri'eqei} \textgreek{to} \textgreek{dendr
\textgreek{H} \textgreek{sugkekrim'enh} \textgreek{'ekdosh} \textgreek{tou} BNF \textgreek{pou} \textgreek


<SQL terminal character> ::=
      <SQL language character>
    | <SQL embedded language character>

<SQL language character> ::=
      <simple Latin letter>
    | <digit>
    | <SQL special character>

<simple Latin letter> ::=
      <simple Latin upper case letter>
    | <simple Latin lower case letter>

<simple Latin upper case letter> ::=
          A | B | C | D | E | F | G | H | I | J | K | L | M | N | O
```

```
    | P | Q | R | S | T | U | V | W | X | Y | Z

<simple Latin lower case letter> ::=
          a | b | c | d | e | f | g | h | i | j | k | l | m | n | o
    | p | q | r | s | t | u | v | w | x | y | z

<digit> ::=
    0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

<SQL special character> ::=
      <space>
    | <double quote>
    | <percent>
    | <ampersand>
    | <quote>
    | <left paren>
    | <right paren>
    | <asterisk>
    | <plus sign>
    | <comma>
    | <minus sign>
    | <period>
    | <solidus>
    | <colon>
    | <semicolon>
    | <less than operator>
    | <equals operator>
    | <greater than operator>
    | <question mark>
    | <underscore>
    | <vertical bar>

<space> ::= !! <EMPHASIS>(space character in character set in use)

<double quote> ::= "

<percent> ::= %

<ampersand> ::= &

<quote> ::= '

<left paren> ::= (

<right paren> ::= )

<asterisk> ::= *

<plus sign> ::= +

<comma> ::= ,

<minus sign> ::= -

<period> ::= .
```

```
<solidus> ::= /

<colon> ::= :

<semicolon> ::= ;

<less than operator> ::= <

<equals operator> ::= =

<greater than operator> ::= >

<question mark> ::= ?

<underscore> ::= _

<vertical bar> ::= |

<SQL embedded language character> ::=
      <left bracket>
    | <right bracket>

<left bracket> ::= [

<right bracket> ::= ]

<token> ::=
      <nondelimiter token>
    | <delimiter token>

<nondelimiter token> ::=
      <regular identifier>
    | <key word>
    | <unsigned numeric literal>
    | <national character string literal>
    | <bit string literal>
    | <hex string literal>

<regular identifier> ::= <identifier body>

<identifier body> ::=
    <identifier start> [ ( <underscore> | <identifier part> )... ]

<identifier start> ::= <EMPHASIS>(!! See the Syntax Rules)

<identifier part> ::=
      <identifier start>
    | <digit>

<key word> ::=
      <reserved word>
    | <non-reserved word>

<reserved word> ::=
```

```
  ABSOLUTE | ACTION | ADD | ALL
| ALLOCATE | ALTER | AND
| ANY | ARE
| AS | ASC
| ASSERTION | AT
| AUTHORIZATION | AVG
| BEGIN | BETWEEN | BIT | BIT_LENGTH
| BOTH | BY
| CASCADE | CASCADED | CASE | CAST
| CATALOG
| CHAR | CHARACTER | CHAR_LENGTH
| CHARACTER_LENGTH | CHECK | CLOSE | COALESCE
| COLLATE | COLLATION
| COLUMN | COMMIT
| CONNECT
| CONNECTION | CONSTRAINT
| CONSTRAINTS | CONTINUE
| CONVERT | CORRESPONDING | COUNT | CREATE | CROSS
| CURRENT
| CURRENT_DATE | CURRENT_TIME
| CURRENT_TIMESTAMP | CURRENT_USER | CURSOR
| DATE | DAY | DEALLOCATE | DEC
| DECIMAL | DECLARE | DEFAULT | DEFERRABLE
| DEFERRED | DELETE | DESC | DESCRIBE | DESCRIPTOR
| DIAGNOSTICS
| DISCONNECT | DISTINCT | DOMAIN | DOUBLE | DROP
| ELSE | END | END-EXEC | ESCAPE
| EXCEPT | EXCEPTION
| EXEC | EXECUTE | EXISTS
| EXTERNAL | EXTRACT
| FALSE | FETCH | FIRST | FLOAT | FOR
| FOREIGN | FOUND | FROM | FULL
| GET | GLOBAL | GO | GOTO
| GRANT | GROUP
| HAVING | HOUR
| IDENTITY | IMMEDIATE | IN | INDICATOR
| INITIALLY | INNER | INPUT
| INSENSITIVE | INSERT | INT | INTEGER | INTERSECT
| INTERVAL | INTO | IS
| ISOLATION
| JOIN
| KEY
| LANGUAGE | LAST | LEADING | LEFT
| LEVEL | LIKE | LOCAL | LOWER
| MATCH | MAX | MIN | MINUTE | MODULE
| MONTH
| NAMES | NATIONAL | NATURAL | NCHAR | NEXT | NO
| NOT | NULL
| NULLIF | NUMERIC
| OCTET_LENGTH | OF
| ON | ONLY | OPEN | OPTION | OR
| ORDER | OUTER
| OUTPUT | OVERLAPS
| PAD | PARTIAL | POSITION | PRECISION | PREPARE
| PRESERVE | PRIMARY
```

```
    | PRIOR | PRIVILEGES | PROCEDURE | PUBLIC
    | READ | REAL | REFERENCES | RELATIVE | RESTRICT
    | REVOKE | RIGHT
    | ROLLBACK | ROWS
    | SCHEMA | SCROLL | SECOND | SECTION
    | SELECT
    | SESSION | SESSION_USER | SET
    | SIZE | SMALLINT | SOME | SPACE | SQL | SQLCODE
    | SQLERROR | SQLSTATE
    | SUBSTRING | SUM | SYSTEM_USER
    | TABLE | TEMPORARY
    | THEN | TIME | TIMESTAMP
    | TIMEZONE_HOUR | TIMEZONE_MINUTE
    | TO | TRAILING | TRANSACTION
    | TRANSLATE | TRANSLATION | TRIM | TRUE
    | UNION | UNIQUE | UNKNOWN | UPDATE | UPPER | USAGE
    | USER | USING
    | VALUE | VALUES | VARCHAR | VARYING | VIEW
    | WHEN | WHENEVER | WHERE | WITH | WORK | WRITE
    | YEAR
    | ZONE

<non-reserved word> ::=

      ADA
    | C | CATALOG_NAME
    | CHARACTER_SET_CATALOG | CHARACTER_SET_NAME
    | CHARACTER_SET_SCHEMA | CLASS_ORIGIN | COBOL | COLLATION_CATALOG
    | COLLATION_NAME | COLLATION_SCHEMA | COLUMN_NAME | COMMAND_FUNCTION
    | COMMITTED
    | CONDITION_NUMBER | CONNECTION_NAME | CONSTRAINT_CATALOG | CONSTRAINT_NAME
    | CONSTRAINT_SCHEMA | CURSOR_NAME
    | DATA | DATETIME_INTERVAL_CODE
    | DATETIME_INTERVAL_PRECISION | DYNAMIC_FUNCTION
    | FORTRAN
    | LENGTH
    | MESSAGE_LENGTH | MESSAGE_OCTET_LENGTH | MESSAGE_TEXT | MORE | MUMPS
    | NAME | NULLABLE | NUMBER
    | PASCAL | PLI
    | REPEATABLE | RETURNED_LENGTH | RETURNED_OCTET_LENGTH | RETURNED_SQLSTATE
    | ROW_COUNT
    | SCALE | SCHEMA_NAME | SERIALIZABLE | SERVER_NAME | SUBCLASS_ORIGIN
    | TABLE_NAME | TYPE
    | UNCOMMITTED | UNNAMED

<unsigned numeric literal> ::=
      <exact numeric literal>
    | <approximate numeric literal>

<exact numeric literal> ::=
      <unsigned integer> [ <period> [ <unsigned integer> ] ]
    | <period> <unsigned integer>

<unsigned integer> ::= <digit>...
```

```
<approximate numeric literal> ::= <mantissa> E <exponent>

<mantissa> ::= <exact numeric literal>

<exponent> ::= <signed integer>

<signed integer> ::= [ <sign> ] <unsigned integer>

<sign> ::= <plus sign> | <minus sign>

<national character string literal> ::=
    N <quote> [ <character representation>... ] <quote>
      [ ( <separator>... <quote> [ <character representation>... ] <quote> )... ]

<character representation> ::=
      <nonquote character>
    | <quote symbol>

<nonquote character> ::= !! <EMPHASIS>(See the Syntax Rules.)

<quote symbol> ::= <quote><quote>

<separator> ::= ( <comment> | <space> | <newline> )...

<comment> ::=
    <comment introducer> [ <comment character>... ] <newline>

<comment introducer> ::= <minus sign><minus sign>[<minus sign>...]

<comment character> ::=
      <nonquote character>
    | <quote>

<newline> ::= !! <EMPHASIS>(implementation-defined end-of-line indicator)

<bit string literal> ::=
    B <quote> [ <bit>... ] <quote>
      [ ( <separator>... <quote> [ <bit>... ] <quote> )... ]

<bit> ::= 0 | 1

<hex string literal> ::=
    X <quote> [ <hexit>... ] <quote>
      [ ( <separator>... <quote> [ <hexit>... ] <quote> )... ]

<hexit> ::= <digit> | A | B | C | D | E | F | a | b | c | d | e | f

<delimiter token> ::=
      <character string literal>
    | <date string>
    | <time string>
    | <timestamp string>
    | <interval string>
    | <delimited identifier>
    | <SQL special character>
```

```
    | <not equals operator>
    | <greater than or equals operator>
    | <less than or equals operator>
    | <concatenation operator>
    | <double period>
    | <left bracket>
    | <right bracket>

<character string literal> ::=
    [ <introducer><character set specification> ]
    <quote> [ <character representation>... ] <quote>
      [ ( <separator>... <quote> [ <character representation>... ] <quote> )... ]

<introducer> ::= <underscore>

<character set specification> ::=
      <standard character repertoire name>
    | <implementation-defined character repertoire name>
    | <user-defined character repertoire name>
    | <standard universal character form-of-use name>
    | <implementation-defined universal character form-of-use name>

<standard character repertoire name> ::= <character set name>

<character set name> ::= [ <schema name> <period> ]
      <SQL language identifier>

<schema name> ::=
    [ <catalog name> <period> ] <unqualified schema name>

<catalog name> ::= <identifier>

<identifier> ::=
    [ <introducer><character set specification> ] <actual identifier>

<actual identifier> ::=
      <regular identifier>
    | <delimited identifier>

<delimited identifier> ::=
    <double quote> <delimited identifier body> <double quote>

<delimited identifier body> ::= <delimited identifier part>...

<delimited identifier part> ::=
      <nondoublequote character>
    | <doublequote symbol>

<nondoublequote character> ::= <EMPHASIS>(!! See the Syntax Rules)

<doublequote symbol> ::= <double quote><double quote>

<unqualified schema name> ::= <identifier>

<SQL language identifier> ::=
```

```
    <SQL language identifier start>
       [ ( <underscore> | <SQL language identifier part> )... ]

<SQL language identifier start> ::= <simple Latin letter>

<SQL language identifier part> ::=
      <simple Latin letter>
    | <digit>

<implementation-defined character repertoire name> ::=
    <character set name>

<user-defined character repertoire name> ::= <character set name>

<standard universal character form-of-use name> ::=
    <character set name>

<implementation-defined universal character form-of-use name> ::=
    <character set name>

<date string> ::=
    <quote> <date value> <quote>

<date value> ::=
    <years value> <minus sign> <months value>
        <minus sign> <days value>

<years value> ::= <datetime value>

<datetime value> ::= <unsigned integer>

<months value> ::= <datetime value>

<days value> ::= <datetime value>

<time string> ::=
    <quote> <time value> [ <time zone interval> ] <quote>

<time value> ::=
    <hours value> <colon> <minutes value> <colon> <seconds value>

<hours value> ::= <datetime value>

<minutes value> ::= <datetime value>

<seconds value> ::=
      <seconds integer value> [ <period> [ <seconds fraction> ] ]

<seconds integer value> ::= <unsigned integer>

<seconds fraction> ::= <unsigned integer>

<time zone interval> ::=
    <sign> <hours value> <colon> <minutes value>
```

```
<timestamp string> ::=
    <quote> <date value> <space> <time value>
        [ <time zone interval> ] <quote>

<interval string> ::=
    <quote> ( <year-month literal> | <day-time literal> ) <quote>

<year-month literal> ::=
      <years value>
    | [ <years value> <minus sign> ] <months value>

<day-time literal> ::=
      <day-time interval>
    | <time interval>

<day-time interval> ::=
    <days value>
      [ <space> <hours value> [ <colon> <minutes value>
        [ <colon> <seconds value> ] ] ]

<time interval> ::=
      <hours value> [ <colon> <minutes value> [ <colon> <seconds value> ] ]
    | <minutes value> [ <colon> <seconds value> ]
    | <seconds value>

<not equals operator> ::= <>

<greater than or equals operator> ::= >=

<less than or equals operator> ::= <=

<concatenation operator> ::= ||

<double period> ::= ..

<module> ::=
    <module name clause>
    <language clause>
    <module authorization clause>
    [ <temporary table declaration>... ]
    <module contents>...

<module name clause> ::=
    MODULE [ <module name> ]
      [ <module character set specification> ]

<module name> ::= <identifier>

<module character set specification> ::=
    NAMES ARE <character set specification>

<language clause> ::=
    LANGUAGE <language name>

<language name> ::=
```

```
    ADA | C | COBOL | FORTRAN | MUMPS | PASCAL | PLI

<module authorization clause> ::=
      SCHEMA <schema name>
    | AUTHORIZATION <module authorization identifier>
    | SCHEMA <schema name>
          AUTHORIZATION <module authorization identifier>

<module authorization identifier> ::=
    <authorization identifier>

<authorization identifier> ::= <identifier>

<temporary table declaration> ::=
    DECLARE LOCAL TEMPORARY TABLE
        <qualified local table name>
      <table element list>
      [ ON COMMIT ( PRESERVE | DELETE ) ROWS ]

<qualified local table name> ::=
    MODULE <period> <local table name>

<local table name> ::= <qualified identifier>

<qualified identifier> ::= <identifier>

<table element list> ::=
      <left paren> <table element> [ ( <comma> <table element> )... ] <right paren>

<table element> ::=
      <column definition>
    | <table constraint definition>

<column definition> ::=
    <column name> ( <data type> | <domain name> )
    [ <default clause> ]
    [ <column constraint definition>... ]
    [ <collate clause> ]

<column name> ::= <identifier>

<data type> ::=
      <character string type>
           [ CHARACTER SET <character set specification> ]
    | <national character string type>
    | <bit string type>
    | <numeric type>
    | <datetime type>
    | <interval type>

<character string type> ::=
      CHARACTER [ <left paren> <length> <right paren> ]
    | CHAR [ <left paren> <length> <right paren> ]
    | CHARACTER VARYING <left paren> <length> <right paren>
    | CHAR VARYING <left paren> <length> <right paren>
```

```
    | VARCHAR <left paren> <length> <right paren>

<length> ::= <unsigned integer>

<national character string type> ::=
      NATIONAL CHARACTER [ <left paren> <length> <right paren> ]
    | NATIONAL CHAR [ <left paren> <length> <right paren> ]
    | NCHAR [ <left paren> <length> <right paren> ]
    | NATIONAL CHARACTER VARYING <left paren> <length> <right paren>
    | NATIONAL CHAR VARYING <left paren> <length> <right paren>
    | NCHAR VARYING <left paren> <length> <right paren>

<bit string type> ::=
      BIT [ <left paren> <length> <right paren> ]
    | BIT VARYING <left paren> <length> <right paren>

<numeric type> ::=
      <exact numeric type>
    | <approximate numeric type>

<exact numeric type> ::=
      NUMERIC [ <left paren> <precision> [ <comma> <scale> ] <right paren> ]
    | DECIMAL [ <left paren> <precision> [ <comma> <scale> ] <right paren> ]
    | DEC [ <left paren> <precision> [ <comma> <scale> ] <right paren> ]
    | INTEGER
    | INT
    | SMALLINT

<precision> ::= <unsigned integer>

<scale> ::= <unsigned integer>

<approximate numeric type> ::=
      FLOAT [ <left paren> <precision> <right paren> ]
    | REAL
    | DOUBLE PRECISION

<datetime type> ::=
      DATE
    | TIME [ <left paren> <time precision> <right paren> ]
          [ WITH TIME ZONE ]
    | TIMESTAMP [ <left paren> <timestamp precision> <right paren> ]
          [ WITH TIME ZONE ]

<time precision> ::= <time fractional seconds precision>

<time fractional seconds precision> ::= <unsigned integer>

<timestamp precision> ::= <time fractional seconds precision>

<interval type> ::= INTERVAL <interval qualifier>

<interval qualifier> ::=
      <start field> TO <end field>
    | <single datetime field>
```

```
<start field> ::=
    <non-second datetime field>
        [ <left paren> <interval leading field precision> <right paren> ]

<non-second datetime field> ::= YEAR | MONTH | DAY | HOUR
    | MINUTE

<interval leading field precision> ::= <unsigned integer>

<end field> ::=
      <non-second datetime field>
    | SECOND [ <left paren> <interval fractional seconds precision> <right paren> ]

<interval fractional seconds precision> ::= <unsigned integer>

<single datetime field> ::=
      <non-second datetime field>
          [ <left paren> <interval leading field precision> <right paren> ]
    | SECOND [ <left paren> <interval leading field precision>
          [ <comma> <interval fractional seconds precision> ] <right paren> ]

<domain name> ::= <qualified name>

<qualified name> ::=
    [ <schema name> <period> ] <qualified identifier>

<default clause> ::=
      DEFAULT <default option>

<default option> ::=
      <literal>
    | <datetime value function>
    | USER
    | CURRENT_USER
    | SESSION_USER
    | SYSTEM_USER
    | NULL

<literal> ::=
      <signed numeric literal>
    | <general literal>

<signed numeric literal> ::=
    [ <sign> ] <unsigned numeric literal>

<general literal> ::=
      <character string literal>
    | <national character string literal>
    | <bit string literal>
    | <hex string literal>
    | <datetime literal>
    | <interval literal>

<datetime literal> ::=
```

```
      <date literal>
    | <time literal>
    | <timestamp literal>

<date literal> ::=
    DATE <date string>

<time literal> ::=
    TIME <time string>

<timestamp literal> ::=
    TIMESTAMP <timestamp string>

<interval literal> ::=
    INTERVAL [ <sign> ] <interval string> <interval qualifier>

<datetime value function> ::=
      <current date value function>
    | <current time value function>
    | <current timestamp value function>

<current date value function> ::= CURRENT_DATE

<current time value function> ::=
      CURRENT_TIME [ <left paren> <time precision> <right paren> ]

<current timestamp value function> ::=
      CURRENT_TIMESTAMP [ <left paren> <timestamp precision> <right paren> ]

<column constraint definition> ::=
    [ <constraint name definition> ]
    <column constraint>
      [ <constraint attributes> ]

<constraint name definition> ::= CONSTRAINT <constraint name>

<constraint name> ::= <qualified name>

<column constraint> ::=
      NOT NULL
    | <unique specification>
    | <references specification>
    | <check constraint definition>

<unique specification> ::=
    UNIQUE | PRIMARY KEY

<references specification> ::=
    REFERENCES <referenced table and columns>
      [ MATCH <match type> ]
      [ <referential triggered action> ]

<referenced table and columns> ::=
     <table name> [ <left paren> <reference column list> <right paren> ]
```

```
<table name> ::=
      <qualified name>
    | <qualified local table name>

<reference column list> ::= <column name list>

<column name list> ::=
    <column name> [ ( <comma> <column name> )... ]

<match type> ::=
      FULL
    | PARTIAL

<referential triggered action> ::=
      <update rule> [ <delete rule> ]
    | <delete rule> [ <update rule> ]

<update rule> ::= ON UPDATE <referential action>

<referential action> ::=
      CASCADE
    | SET NULL
    | SET DEFAULT
    | NO ACTION

<delete rule> ::= ON DELETE <referential action>

<check constraint definition> ::=
    CHECK
        <left paren> <search condition> <right paren>

<search condition> ::=
      <boolean term>
    | <search condition> OR <boolean term>

<boolean term> ::=
      <boolean factor>
    | <boolean term> AND <boolean factor>

<boolean factor> ::=
    [ NOT ] <boolean test>

<boolean test> ::=
    <boolean primary> [ IS [ NOT ]
          <truth value> ]

<boolean primary> ::=
      <predicate>
    | <left paren> <search condition> <right paren>

<predicate> ::=
      <comparison predicate>
    | <between predicate>
    | <in predicate>
    | <like predicate>
```

```
    | <null predicate>
    | <quantified comparison predicate>
    | <exists predicate>
    | <unique predicate>
    | <match predicate>
    | <overlaps predicate>

<comparison predicate> ::=
    <row value constructor> <comp op>
        <row value constructor>

<row value constructor> ::=
       <row value constructor element>
    | <left paren> <row value constructor list> <right paren>
    | <row subquery>

<row value constructor element> ::=
      <value expression>
    | <null specification>
    | <default specification>

<value expression> ::=
      <numeric value expression>
    | <string value expression>
    | <datetime value expression>
    | <interval value expression>

<numeric value expression> ::=
      <term>
    | <numeric value expression> <plus sign> <term>
    | <numeric value expression> <minus sign> <term>

<term> ::=
      <factor>
    | <term> <asterisk> <factor>
    | <term> <solidus> <factor>

<factor> ::=
    [ <sign> ] <numeric primary>

<numeric primary> ::=
      <value expression primary>
    | <numeric value function>

<value expression primary> ::=
      <unsigned value specification>
    | <column reference>
    | <set function specification>
    | <scalar subquery>
    | <case expression>
    | <left paren> <value expression> <right paren>
    | <cast specification>

<unsigned value specification> ::=
      <unsigned literal>
```

```
    | <general value specification>

<unsigned literal> ::=
      <unsigned numeric literal>
    | <general literal>

<general value specification> ::=
      <parameter specification>
    | <dynamic parameter specification>
    | <variable specification>
    | USER
    | CURRENT_USER
    | SESSION_USER
    | SYSTEM_USER
    | VALUE

<parameter specification> ::=
    <parameter name> [ <indicator parameter> ]

<parameter name> ::= <colon> <identifier>

<indicator parameter> ::=
    [ INDICATOR ] <parameter name>

<dynamic parameter specification> ::= <question mark>

<variable specification> ::=
    <embedded variable name> [ <indicator variable> ]

<embedded variable name> ::=
    <colon><host identifier>

<host identifier> ::=
      <Ada host identifier>
    | <C host identifier>
    | <COBOL host identifier>
    | <Fortran host identifier>
    | <MUMPS host identifier>
    | <Pascal host identifier>
    | <PL/I host identifier>

<Ada host identifier> ::= !! <EMPHASIS>(See the Syntax Rules.)

<C host identifier> ::=
    !! <EMPHASIS>(See the Syntax Rules.)

<COBOL host identifier> ::= !! <EMPHASIS>(See the Syntax Rules.)

<Fortran host identifier> ::= !! <EMPHASIS>(See the Syntax Rules.)

<MUMPS host identifier> ::= !! <EMPHASIS>(See the Syntax Rules.)

<Pascal host identifier> ::= !! <EMPHASIS>(See the Syntax Rules.)

<PL/I host identifier> ::= !! <EMPHASIS>(See the Syntax Rules.)
```

```
<indicator variable> ::=
    [ INDICATOR ] <embedded variable name>

<column reference> ::= [ <qualifier> <period> ] <column name>

<qualifier> ::=
      <table name>
    | <correlation name>

<correlation name> ::= <identifier>

<set function specification> ::=
      COUNT <left paren> <asterisk> <right paren>
    | <general set function>

<general set function> ::=
      <set function type>
          <left paren> [ <set quantifier> ] <value expression> <right paren>

<set function type> ::=
    AVG | MAX | MIN | SUM | COUNT

<set quantifier> ::= DISTINCT | ALL

<scalar subquery> ::= <subquery>

<subquery> ::= <left paren> <query expression> <right paren>

<query expression> ::=
      <non-join query expression>
    | <joined table>

<non-join query expression> ::=
      <non-join query term>
    | <query expression> UNION  [ ALL ]
          [ <corresponding spec> ] <query term>
    | <query expression> EXCEPT [ ALL ]
          [ <corresponding spec> ] <query term>

<non-join query term> ::=
      <non-join query primary>
    | <query term> INTERSECT [ ALL ]
          [ <corresponding spec> ] <query primary>

<non-join query primary> ::=
      <simple table>
    | <left paren> <non-join query expression> <right paren>

<simple table> ::=
      <query specification>
    | <table value constructor>
    | <explicit table>

<query specification> ::=
```

```
    SELECT [ <set quantifier> ] <select list> <table expression>

<select list> ::=
      <asterisk>
    | <select sublist> [ ( <comma> <select sublist> )... ]

<select sublist> ::=
      <derived column>
    | <qualifier> <period> <asterisk>

<derived column> ::= <value expression> [ <as clause> ]

<as clause> ::= [ AS ] <column name>

<table expression> ::=
    <from clause>
    [ <where clause> ]
    [ <group by clause> ]
    [ <having clause> ]

<from clause> ::= FROM <table reference>
    [ ( <comma> <table reference> )... ]

<table reference> ::=
      <table name> [ [ AS ] <correlation name>
          [ <left paren> <derived column list> <right paren> ] ]
    | <derived table> [ AS ] <correlation name>
          [ <left paren> <derived column list> <right paren> ]
    | <joined table>

<derived column list> ::= <column name list>

<derived table> ::= <table subquery>

<table subquery> ::= <subquery>

<joined table> ::=
      <cross join>
    | <qualified join>
    | <left paren> <joined table> <right paren>

<cross join> ::=
    <table reference> CROSS JOIN <table reference>

<qualified join> ::=
    <table reference> [ NATURAL ] [ <join type> ] JOIN
      <table reference> [ <join specification> ]

<join type> ::=
      INNER
    | <outer join type> [ OUTER ]
    | UNION

<outer join type> ::=
      LEFT
```

```
    | RIGHT
    | FULL

<join specification> ::=
      <join condition>
    | <named columns join>

<join condition> ::= ON <search condition>

<named columns join> ::=
    USING <left paren> <join column list> <right paren>

<join column list> ::= <column name list>

<where clause> ::= WHERE <search condition>

<group by clause> ::=
    GROUP BY <grouping column reference list>

<grouping column reference list> ::=
    <grouping column reference>
        [ ( <comma> <grouping column reference> )... ]

<grouping column reference> ::=
    <column reference> [ <collate clause> ]

<collate clause> ::= COLLATE <collation name>

<collation name> ::= <qualified name>

<having clause> ::= HAVING <search condition>

<table value constructor> ::=
    VALUES <table value constructor list>

<table value constructor list> ::=
    <row value constructor> [ ( <comma> <row value constructor> )... ]

<explicit table> ::= TABLE <table name>

<query term> ::=
      <non-join query term>
    | <joined table>

<corresponding spec> ::=
    CORRESPONDING [ BY <left paren> <corresponding column list> <right paren> ]

<corresponding column list> ::= <column name list>

<query primary> ::=
      <non-join query primary>
    | <joined table>

<case expression> ::=
      <case abbreviation>
```

```
    | <case specification>

<case abbreviation> ::=
      NULLIF <left paren> <value expression> <comma>
            <value expression> <right paren>
    | COALESCE <left paren> <value expression>
            ( <comma> <value expression> )... <right paren>

<case specification> ::=
      <simple case>
    | <searched case>

<simple case> ::=
    CASE <case operand>
      <simple when clause>...
      [ <else clause> ]
    END

<case operand> ::= <value expression>

<simple when clause> ::= WHEN <when operand> THEN <result>

<when operand> ::= <value expression>

<result> ::= <result expression> | NULL

<result expression> ::= <value expression>

<else clause> ::= ELSE <result>

<searched case> ::=
    CASE
      <searched when clause>...
      [ <else clause> ]
    END

<searched when clause> ::= WHEN <search condition> THEN <result>

<cast specification> ::=
    CAST <left paren> <cast operand> AS
        <cast target> <right paren>

<cast operand> ::=
      <value expression>
    | NULL

<cast target> ::=
      <domain name>
    | <data type>

<numeric value function> ::=
      <position expression>
    | <extract expression>
    | <length expression>
```

```
<position expression> ::=
    POSITION <left paren> <character value expression>
        IN <character value expression> <right paren>

<character value expression> ::=
      <concatenation>
    | <character factor>

<concatenation> ::=
    <character value expression> <concatenation operator>
        <character factor>

<character factor> ::=
    <character primary> [ <collate clause> ]

<character primary> ::=
      <value expression primary>
    | <string value function>

<string value function> ::=
      <character value function>
    | <bit value function>

<character value function> ::=
      <character substring function>
    | <fold>
    | <form-of-use conversion>
    | <character translation>
    | <trim function>

<character substring function> ::=
    SUBSTRING <left paren> <character value expression> FROM <start position>
                [ FOR <string length> ] <right paren>

<start position> ::= <numeric value expression>

<string length> ::= <numeric value expression>

<fold> ::= ( UPPER | LOWER )
     <left paren> <character value expression> <right paren>

<form-of-use conversion> ::=
    CONVERT <left paren> <character value expression>
        USING <form-of-use conversion name> <right paren>

<form-of-use conversion name> ::= <qualified name>

<character translation> ::=
    TRANSLATE <left paren> <character value expression>
        USING <translation name> <right paren>

<translation name> ::= <qualified name>

<trim function> ::=
    TRIM <left paren> <trim operands> <right paren>
```

```
<trim operands> ::=
    [ [ <trim specification> ] [ <trim character> ] FROM ] <trim source>

<trim specification> ::=
      LEADING
    | TRAILING
    | BOTH

<trim character> ::= <character value expression>

<trim source> ::= <character value expression>

<bit value function> ::=
    <bit substring function>

<bit substring function> ::=
    SUBSTRING <left paren> <bit value expression> FROM <start position>
        [ FOR <string length> ] <right paren>

<bit value expression> ::=
      <bit concatenation>
    | <bit factor>

<bit concatenation> ::=
    <bit value expression> <concatenation operator> <bit factor>

<bit factor> ::= <bit primary>

<bit primary> ::=
      <value expression primary>
    | <string value function>

<extract expression> ::=
    EXTRACT <left paren> <extract field>
        FROM <extract source> <right paren>

<extract field> ::=
      <datetime field>
    | <time zone field>

<datetime field> ::=
      <non-second datetime field>
    | SECOND

<time zone field> ::=
      TIMEZONE_HOUR
    | TIMEZONE_MINUTE

<extract source> ::=
      <datetime value expression>
    | <interval value expression>

<datetime value expression> ::=
      <datetime term>
```

```
    | <interval value expression> <plus sign> <datetime term>
    | <datetime value expression> <plus sign> <interval term>
    | <datetime value expression> <minus sign> <interval term>

<interval term> ::=
      <interval factor>
    | <interval term 2> <asterisk> <factor>
    | <interval term 2> <solidus> <factor>
    | <term> <asterisk> <interval factor>

<interval factor> ::=
    [ <sign> ] <interval primary>

<interval primary> ::=
      <value expression primary> [ <interval qualifier> ]

<interval term 2> ::= <interval term>

<interval value expression> ::=
      <interval term>
    | <interval value expression 1> <plus sign> <interval term 1>
    | <interval value expression 1> <minus sign> <interval term 1>
    | <left paren> <datetime value expression> <minus sign>
          <datetime term> <right paren> <interval qualifier>

<interval value expression 1> ::= <interval value expression>

<interval term 1> ::= <interval term>

<datetime term> ::=
      <datetime factor>

<datetime factor> ::=
      <datetime primary> [ <time zone> ]

<datetime primary> ::=
      <value expression primary>
    | <datetime value function>

<time zone> ::=
    AT <time zone specifier>

<time zone specifier> ::=
      LOCAL
    | TIME ZONE <interval value expression>

<length expression> ::=
      <char length expression>
    | <octet length expression>
    | <bit length expression>

<char length expression> ::=
    ( CHAR_LENGTH | CHARACTER_LENGTH )
        <left paren> <string value expression> <right paren>
```

```
<string value expression> ::=
      <character value expression>
    | <bit value expression>

<octet length expression> ::=
    OCTET_LENGTH <left paren> <string value expression> <right paren>

<bit length expression> ::=
    BIT_LENGTH <left paren> <string value expression> <right paren>

<null specification> ::=
    NULL

<default specification> ::=
    DEFAULT

<row value constructor list> ::=
    <row value constructor element>
        [ ( <comma> <row value constructor element> )... ]

<row subquery> ::= <subquery>

<comp op> ::=
      <equals operator>
    | <not equals operator>
    | <less than operator>
    | <greater than operator>
    | <less than or equals operator>
    | <greater than or equals operator>

<between predicate> ::=
    <row value constructor> [ NOT ] BETWEEN
      <row value constructor> AND <row value constructor>

<in predicate> ::=
    <row value constructor>
      [ NOT ] IN <in predicate value>

<in predicate value> ::=
      <table subquery>
    | <left paren> <in value list> <right paren>

<in value list> ::=
    <value expression> ( <comma> <value expression> )...

<like predicate> ::=
    <match value> [ NOT ] LIKE <pattern>
      [ ESCAPE <escape character> ]

<match value> ::= <character value expression>

<pattern> ::= <character value expression>

<escape character> ::= <character value expression>
```

```
<null predicate> ::= <row value constructor>
    IS [ NOT ] NULL

<quantified comparison predicate> ::=
    <row value constructor> <comp op> <quantifier> <table subquery>

<quantifier> ::= <all> | <some>

<all> ::= ALL

<some> ::= SOME | ANY

<exists predicate> ::= EXISTS <table subquery>

<unique predicate> ::= UNIQUE <table subquery>

<match predicate> ::=
    <row value constructor> MATCH [ UNIQUE ]
        [ PARTIAL | FULL ] <table subquery>

<overlaps predicate> ::=
    <row value constructor 1> OVERLAPS <row value constructor 2>

<row value constructor 1> ::= <row value constructor>

<row value constructor 2> ::= <row value constructor>

<truth value> ::=
      TRUE
    | FALSE
    | UNKNOWN

<constraint attributes> ::=
      <constraint check time> [ [ NOT ] DEFERRABLE ]
    | [ NOT ] DEFERRABLE [ <constraint check time> ]

<constraint check time> ::=
      INITIALLY DEFERRED
    | INITIALLY IMMEDIATE

<table constraint definition> ::=
    [ <constraint name definition> ]
    <table constraint> [ <constraint attributes> ]

<table constraint> ::=
      <unique constraint definition>
    | <referential constraint definition>
    | <check constraint definition>

<unique constraint definition> ::=
            <unique specification> even in SQL3)
    <unique specification>
      <left paren> <unique column list> <right paren>

<unique column list> ::= <column name list>
```

```
<referential constraint definition> ::=
    FOREIGN KEY
        <left paren> <referencing columns> <right paren>
      <references specification>

<referencing columns> ::=
    <reference column list>

<module contents> ::=
      <declare cursor>
    | <dynamic declare cursor>
    | <procedure>

<declare cursor> ::=
    DECLARE <cursor name> [ INSENSITIVE ] [ SCROLL ] CURSOR
      FOR <cursor specification>

<cursor name> ::= <identifier>

<cursor specification> ::=
    <query expression> [ <order by clause> ]
      [ <updatability clause> ]

<order by clause> ::=
    ORDER BY <sort specification list>

<sort specification list> ::=
    <sort specification> [ ( <comma> <sort specification> )... ]

<sort specification> ::=
    <sort key> [ <collate clause> ] [ <ordering specification> ]

<sort key> ::=
      <column name>
    | <unsigned integer>

<ordering specification> ::= ASC | DESC

<updatability clause> ::=
    FOR
        ( READ ONLY |
          UPDATE [ OF <column name list> ] )

<dynamic declare cursor> ::=
    DECLARE <cursor name> [ INSENSITIVE ] [ SCROLL ] CURSOR
        FOR <statement name>

<statement name> ::= <identifier>

<procedure> ::=
    PROCEDURE <procedure name>
        <parameter declaration list> <semicolon>
      <SQL procedure statement> <semicolon>
```

```
<procedure name> ::= <identifier>

<parameter declaration list> ::=
      <left paren> <parameter declaration>
          [ ( <comma> <parameter declaration> )... ] <right paren>
    | <parameter declaration>...

<parameter declaration> ::=
      <parameter name> <data type>
    | <status parameter>

<status parameter> ::=
    SQLCODE | SQLSTATE

<SQL procedure statement> ::=
      <SQL schema statement>
    | <SQL data statement>
    | <SQL transaction statement>
    | <SQL connection statement>
    | <SQL session statement>
    | <SQL dynamic statement>
    | <SQL diagnostics statement>

<SQL schema statement> ::=
      <SQL schema definition statement>
    | <SQL schema manipulation statement>

<SQL schema definition statement> ::=
      <schema definition>
    | <table definition>
    | <view definition>
    | <grant statement>
    | <domain definition>
    | <character set definition>
    | <collation definition>
    | <translation definition>
    | <assertion definition>

<schema definition> ::=
    CREATE SCHEMA <schema name clause>
      [ <schema character set specification> ]
      [ <schema element>... ]

<schema name clause> ::=
      <schema name>
    | AUTHORIZATION <schema authorization identifier>
    | <schema name> AUTHORIZATION
          <schema authorization identifier>

<schema authorization identifier> ::=
    <authorization identifier>

<schema character set specification> ::=
    DEFAULT CHARACTER
        SET <character set specification>
```

```
<schema element> ::=
      <domain definition>
    | <table definition>
    | <view definition>
    | <grant statement>
    | <assertion definition>
    | <character set definition>
    | <collation definition>
    | <translation definition>

<domain definition> ::=
    CREATE DOMAIN <domain name>
        [ AS ] <data type>
      [ <default clause> ]
      [ <domain constraint>... ]
      [ <collate clause> ]

<domain constraint> ::=
    [ <constraint name definition> ]
    <check constraint definition> [ <constraint attributes> ]

<table definition> ::=
    CREATE [ ( GLOBAL | LOCAL ) TEMPORARY ] TABLE
        <table name>
      <table element list>
      [ ON COMMIT ( DELETE | PRESERVE ) ROWS ]

<view definition> ::=
    CREATE VIEW <table name> [ <left paren> <view column list>
                             <right paren> ]
      AS <query expression>
      [ WITH [ <levels clause> ] CHECK OPTION ]

<view column list> ::= <column name list>

<levels clause> ::=
    CASCADED | LOCAL

<grant statement> ::=
   GRANT <privileges> ON <object name>
     TO <grantee> [ ( <comma> <grantee> )... ]
       [ WITH GRANT OPTION ]

<privileges> ::=
      ALL PRIVILEGES
    | <action list>

<action list> ::= <action> [ ( <comma> <action> )... ]

<action> ::=
      SELECT
    | DELETE
    | INSERT [ <left paren> <privilege column list> <right paren> ]
    | UPDATE [ <left paren> <privilege column list> <right paren> ]
```

```
    | REFERENCES [ <left paren> <privilege column list> <right paren> ]
    | USAGE

<privilege column list> ::= <column name list>

<object name> ::=
      [ TABLE ] <table name>
    | DOMAIN <domain name>
    | COLLATION <collation name>
    | CHARACTER SET <character set name>
    | TRANSLATION <translation name>

<grantee> ::=
      PUBLIC
    | <authorization identifier>

<assertion definition> ::=
    CREATE ASSERTION <constraint name> <assertion check>
      [ <constraint attributes> ]

<assertion check> ::=
    CHECK
        <left paren> <search condition> <right paren>

<character set definition> ::=
    CREATE CHARACTER SET <character set name>
        [ AS ]
      <character set source>
      [ <collate clause> | <limited collation definition> ]

<character set source> ::=
      GET <existing character set name>

<existing character set name> ::=
      <standard character repertoire name>
    | <implementation-defined character repertoire name>
    | <schema character set name>

<schema character set name> ::= <character set name>

<limited collation definition> ::=
    COLLATION FROM <collation source>

<collation source> ::=
      <collating sequence definition>
    | <translation collation>

<collating sequence definition> ::=
      <external collation>
    | <schema collation name>
    | DESC <left paren> <collation name> <right paren>
    | DEFAULT

<external collation> ::=
    EXTERNAL <left paren> <quote> <external collation name> <quote> <right paren>
```

```
<external collation name> ::=
      <standard collation name>
    | <implementation-defined collation name>

<standard collation name> ::= <collation name>

<implementation-defined collation name> ::= <collation name>

<schema collation name> ::= <collation name>

<translation collation> ::=
    TRANSLATION <translation name>
        [ THEN COLLATION <collation name> ]

<collation definition> ::=
    CREATE COLLATION <collation name> FOR
        <character set specification>
      FROM <collation source>
        [ <pad attribute> ]

<pad attribute> ::=
      NO PAD
    | PAD SPACE

<translation definition> ::=
    CREATE TRANSLATION <translation name>
      FOR <source character set specification>
        TO <target character set specification>
      FROM <translation source>

<source character set specification> ::= <character set specification>

<target character set specification> ::= <character set specification>

<translation source> ::=
      <translation specification>

<translation specification> ::=
      <external translation>
    | IDENTITY
    | <schema translation name>

<external translation> ::=
    EXTERNAL <left paren> <quote> <external translation name> <quote> <right paren>

<external translation name> ::=
      <standard translation name>
    | <implementation-defined translation name>

<standard translation name> ::= <translation name>

<implementation-defined translation name> ::= <translation name>

<schema translation name> ::= <translation name>
```

```
<SQL schema manipulation statement> ::=
      <drop schema statement>
    | <alter table statement>
    | <drop table statement>
    | <drop view statement>
    | <revoke statement>
    | <alter domain statement>
    | <drop domain statement>
    | <drop character set statement>
    | <drop collation statement>
    | <drop translation statement>
    | <drop assertion statement>

<drop schema statement> ::=
    DROP SCHEMA <schema name> <drop behavior>

<drop behavior> ::= CASCADE | RESTRICT

<alter table statement> ::=
    ALTER TABLE <table name> <alter table action>

<alter table action> ::=
      <add column definition>
    | <alter column definition>
    | <drop column definition>
    | <add table constraint definition>
    | <drop table constraint definition>

<add column definition> ::=
    ADD [ COLUMN ] <column definition>

<alter column definition> ::=
    ALTER [ COLUMN ] <column name> <alter column action>

<alter column action> ::=
      <set column default clause>
    | <drop column default clause>

<set column default clause> ::=
    SET <default clause>

<drop column default clause> ::=
    DROP DEFAULT

<drop column definition> ::=
    DROP [ COLUMN ] <column name> <drop behavior>

<add table constraint definition> ::=
    ADD <table constraint definition>

<drop table constraint definition> ::=
    DROP CONSTRAINT <constraint name> <drop behavior>

<drop table statement> ::=
```

```
    DROP TABLE <table name> <drop behavior>

<drop view statement> ::=
    DROP VIEW <table name> <drop behavior>

<revoke statement> ::=
    REVOKE [ GRANT OPTION FOR ]
        <privileges>
        ON <object name>
      FROM <grantee> [ ( <comma> <grantee> )... ] <drop behavior>

<alter domain statement> ::=
    ALTER DOMAIN <domain name> <alter domain action>

<alter domain action> ::=
      <set domain default clause>
    | <drop domain default clause>
    | <add domain constraint definition>
    | <drop domain constraint definition>

<set domain default clause> ::= SET <default clause>

<drop domain default clause> ::= DROP DEFAULT

<add domain constraint definition> ::=
    ADD <domain constraint>

<drop domain constraint definition> ::=
    DROP CONSTRAINT <constraint name>

<drop domain statement> ::=
    DROP DOMAIN <domain name> <drop behavior>

<drop character set statement> ::=
    DROP CHARACTER SET <character set name>

<drop collation statement> ::=
    DROP COLLATION <collation name>

<drop translation statement> ::=
    DROP TRANSLATION <translation name>

<drop assertion statement> ::=
    DROP ASSERTION <constraint name>

<SQL data statement> ::=
      <open statement>
    | <fetch statement>
    | <close statement>
    | <select statement: single row>
    | <SQL data change statement>

<open statement> ::=
    OPEN <cursor name>
```

```
<fetch statement> ::=
    FETCH [ [ <fetch orientation> ] FROM ]
      <cursor name> INTO <fetch target list>

<fetch orientation> ::=
      NEXT
    | PRIOR
    | FIRST
    | LAST
    | ( ABSOLUTE | RELATIVE ) <simple value specification>

<simple value specification> ::=
      <parameter name>
    | <embedded variable name>
    | <literal>

<fetch target list> ::=
    <target specification> [ ( <comma> <target specification> )... ]

<target specification> ::=
      <parameter specification>
    | <variable specification>

<close statement> ::=
    CLOSE <cursor name>

<select statement: single row> ::=
    SELECT [ <set quantifier> ] <select list>
      INTO <select target list>
        <table expression>

<select target list> ::=
    <target specification> [ ( <comma> <target specification> )... ]

<SQL data change statement> ::=
      <delete statement: positioned>
    | <delete statement: searched>
    | <insert statement>
    | <update statement: positioned>
    | <update statement: searched>

<delete statement: positioned> ::=
    DELETE FROM <table name>
      WHERE CURRENT OF <cursor name>

<delete statement: searched> ::=
    DELETE FROM <table name>
      [ WHERE <search condition> ]

<insert statement> ::=
    INSERT INTO <table name>
      <insert columns and source>

<insert columns and source> ::=
      [ <left paren> <insert column list> <right paren> ]
```

```
            <query expression>
      | DEFAULT VALUES

<insert column list> ::= <column name list>

<update statement: positioned> ::=
    UPDATE <table name>
      SET <set clause list>
        WHERE CURRENT OF <cursor name>

<set clause list> ::=
    <set clause> [ ( <comma> <set clause> )... ]

<set clause> ::=
    <object column> <equals operator> <update source>

<object column> ::= <column name>

<update source> ::=
      <value expression>
    | <null specification>
    | DEFAULT

<update statement: searched> ::=
    UPDATE <table name>
      SET <set clause list>
      [ WHERE <search condition> ]

<SQL transaction statement> ::=
      <set transaction statement>
    | <set constraints mode statement>
    | <commit statement>
    | <rollback statement>

<set transaction statement> ::=
    SET TRANSACTION <transaction mode>
        [ ( <comma> <transaction mode> )... ]

<transaction mode> ::=
      <isolation level>
    | <transaction access mode>
    | <diagnostics size>

<isolation level> ::=
    ISOLATION LEVEL <level of isolation>

<level of isolation> ::=
      READ UNCOMMITTED
    | READ COMMITTED
    | REPEATABLE READ
    | SERIALIZABLE

<transaction access mode> ::=
      READ ONLY
    | READ WRITE
```

```
<diagnostics size> ::=
    DIAGNOSTICS SIZE <number of conditions>

<number of conditions> ::= <simple value specification>

<set constraints mode statement> ::=
    SET CONSTRAINTS <constraint name list>
        ( DEFERRED | IMMEDIATE )

<constraint name list> ::=
      ALL
    | <constraint name> [ ( <comma> <constraint name> )... ]

<commit statement> ::=
    COMMIT [ WORK ]

<rollback statement> ::=
    ROLLBACK [ WORK ]

<SQL connection statement> ::=
      <connect statement>
    | <set connection statement>
    | <disconnect statement>

<connect statement> ::=
    CONNECT TO <connection target>

<connection target> ::=
      <SQL-server name>
        [ AS <connection name> ]
          correspondence with Tony Gordon)
        [ USER <user name> ]
    | DEFAULT

<SQL-server name> ::= <simple value specification>

<connection name> ::= <simple value specification>

<user name> ::= <simple value specification>

<set connection statement> ::=
    SET CONNECTION <connection object>

<connection object> ::=
      DEFAULT
    | <connection name>

<disconnect statement> ::=
    DISCONNECT <disconnect object>

<disconnect object> ::=
      <connection object>
    | ALL
    | CURRENT
```

```
<SQL session statement> ::=
      <set catalog statement>
    | <set schema statement>
    | <set names statement>
    | <set session authorization identifier statement>
    | <set local time zone statement>

<set catalog statement> ::=
    SET CATALOG <value specification>

<value specification> ::=
      <literal>
    | <general value specification>

<set schema statement> ::=
    SET SCHEMA <value specification>

<set names statement> ::=
    SET NAMES <value specification>

<set session authorization identifier statement> ::=
    SET SESSION AUTHORIZATION
        <value specification>

<set local time zone statement> ::=
    SET TIME ZONE
        <set time zone value>

<set time zone value> ::=
      <interval value expression>
    | LOCAL

<SQL dynamic statement> ::=
      <system descriptor statement>
    | <prepare statement>
    | <deallocate prepared statement>
    | <describe statement>
    | <execute statement>
    | <execute immediate statement>
    | <SQL dynamic data statement>

<system descriptor statement> ::=
      <allocate descriptor statement>
    | <deallocate descriptor statement>
    | <set descriptor statement>
    | <get descriptor statement>

<allocate descriptor statement> ::=
    ALLOCATE DESCRIPTOR <descriptor name>
       [ WITH MAX <occurrences> ]

<descriptor name> ::=
    [ <scope option> ] <simple value specification>
```

```
<scope option> ::=
      GLOBAL
    | LOCAL

<occurrences> ::= <simple value specification>

<deallocate descriptor statement> ::=
    DEALLOCATE DESCRIPTOR <descriptor name>

<set descriptor statement> ::=
    SET DESCRIPTOR <descriptor name>
        <set descriptor information>

<set descriptor information> ::=
      <set count>
    | VALUE <item number>
        <set item information> [ ( <comma> <set item information> )... ]

<set count> ::=
    COUNT <equals operator> <simple value specification 1>

<simple value specification 1> ::= <simple value specification>

<item number> ::= <simple value specification>

<set item information> ::=
    <descriptor item name> <equals operator> <simple value specification 2>

<descriptor item name> ::=
      TYPE
    | LENGTH
    | OCTET_LENGTH
    | RETURNED_LENGTH
    | RETURNED_OCTET_LENGTH
    | PRECISION
    | SCALE
    | DATETIME_INTERVAL_CODE
    | DATETIME_INTERVAL_PRECISION
    | NULLABLE
    | INDICATOR
    | DATA
    | NAME
    | UNNAMED
    | COLLATION_CATALOG
    | COLLATION_SCHEMA
    | COLLATION_NAME
    | CHARACTER_SET_CATALOG
    | CHARACTER_SET_SCHEMA
    | CHARACTER_SET_NAME

<simple value specification 2> ::= <simple value specification>

<item number> ::= <simple value specification>

<get descriptor statement> ::=
```

```
    GET DESCRIPTOR <descriptor name> <get descriptor information>

<get descriptor information> ::=
      <get count>
    | VALUE <item number>
        <get item information> [ ( <comma> <get item information> )... ]

<get count> ::=
    <simple target specification 1> <equals operator>
         COUNT

<simple target specification 1> ::= <simple target specification>

<simple target specification> ::=
      <parameter name>
    | <embedded variable name>

<get item information> ::=
    <simple target specification 2> <equals operator> <descriptor item name>>

<simple target specification 2> ::= <simple target specification>

<prepare statement> ::=
    PREPARE <SQL statement name> FROM <SQL statement variable>

<SQL statement name> ::=
      <statement name>
    | <extended statement name>

<extended statement name> ::=
    [ <scope option> ] <simple value specification>

<SQL statement variable> ::= <simple value specification>

<deallocate prepared statement> ::=
    DEALLOCATE PREPARE <SQL statement name>

<describe statement> ::=
      <describe input statement>
    | <describe output statement>

<describe input statement> ::=
    DESCRIBE INPUT <SQL statement name> <using descriptor>

<using descriptor> ::=
    ( USING | INTO ) SQL DESCRIPTOR <descriptor name>

<describe output statement> ::=
    DESCRIBE [ OUTPUT ] <SQL statement name> <using descriptor>

<execute statement> ::=
    EXECUTE <SQL statement name>
      [ <result using clause> ]
      [ <parameter using clause> ]
```

```
<result using clause> ::= <using clause>

<using clause> ::=
      <using arguments>
    | <using descriptor>

<using arguments> ::=
    ( USING | INTO ) <argument> [ ( <comma> <argument> )... ]

<argument> ::= <target specification>

<parameter using clause> ::= <using clause>

<execute immediate statement> ::=
    EXECUTE IMMEDIATE <SQL statement variable>

<SQL dynamic data statement> ::=
      <allocate cursor statement>
    | <dynamic open statement>
    | <dynamic fetch statement>
    | <dynamic close statement>
    | <dynamic delete statement: positioned>
    | <dynamic update statement: positioned>

<allocate cursor statement> ::=
    ALLOCATE <extended cursor name> [ INSENSITIVE ]
        [ SCROLL ] CURSOR
      FOR <extended statement name>

<extended cursor name> ::=
    [ <scope option> ] <simple value specification>

<dynamic open statement> ::=
    OPEN <dynamic cursor name> [ <using clause> ]

<dynamic cursor name> ::=
      <cursor name>
    | <extended cursor name>

<dynamic fetch statement> ::=
    FETCH [ [ <fetch orientation> ] FROM ] <dynamic cursor name>
        <using clause>

<dynamic close statement> ::=
    CLOSE <dynamic cursor name>

<dynamic delete statement: positioned> ::=
    DELETE FROM <table name>
      WHERE CURRENT OF
          <dynamic cursor name>

<dynamic update statement: positioned> ::=
    UPDATE <table name>
      SET <set clause>
          [ ( <comma> <set clause> )... ]
```

```
        WHERE CURRENT OF
            <dynamic cursor name>

<SQL diagnostics statement> ::=
    <get diagnostics statement>

<get diagnostics statement> ::=
    GET DIAGNOSTICS <sql diagnostics information>

<sql diagnostics information> ::=
      <statement information>
    | <condition information>

<statement information> ::=
    <statement information item> [ ( <comma> <statement information item> )... ]

<statement information item> ::=
    <simple target specification> <equals operator> <statement information item name>

<statement information item name> ::=
      NUMBER
    | MORE
    | COMMAND_FUNCTION
    | DYNAMIC_FUNCTION
    | ROW_COUNT

<condition information> ::=
    EXCEPTION <condition number>
      <condition information item> [ ( <comma> <condition information item> )... ]

<condition number> ::= <simple value specification>

<condition information item> ::=
    <simple target specification> <equals operator> <condition information item name>

<condition information item name> ::=
      CONDITION_NUMBER
    | RETURNED_SQLSTATE
    | CLASS_ORIGIN
    | SUBCLASS_ORIGIN
    | SERVER_NAME
    | CONNECTION_NAME
    | CONSTRAINT_CATALOG
    | CONSTRAINT_SCHEMA
    | CONSTRAINT_NAME
    | CATALOG_NAME
    | SCHEMA_NAME
    | TABLE_NAME
    | COLUMN_NAME
    | CURSOR_NAME
    | MESSAGE_TEXT
    | MESSAGE_LENGTH
    | MESSAGE_OCTET_LENGTH

<embedded SQL host program> ::=
```

```
      <embedded SQL Ada program>
    | <embedded SQL C program>
    | <embedded SQL COBOL program>
    | <embedded SQL Fortran program>
    | <embedded SQL MUMPS program>
    | <embedded SQL Pascal program>
    | <embedded SQL PL/I program>

<embedded SQL Ada program> ::= !! <EMPHASIS>(See the Syntax Rules.)

<embedded SQL C program> ::=
      !! <EMPHASIS>(See the Syntax Rules.)

<embedded SQL COBOL program> ::= !! <EMPHASIS>(See the Syntax Rules.)

<embedded SQL Fortran program> ::=
    !! <EMPHASIS>(See the Syntax Rules.)

<embedded SQL MUMPS program> ::= !! <EMPHASIS>(See the Syntax Rules.)

<embedded SQL Pascal program> ::=
    !! <EMPHASIS>(See the Syntax Rules.)

<embedded SQL PL/I program> ::= !! <EMPHASIS>(See the Syntax Rules.)

<embedded SQL declare section> ::=
      <embedded SQL begin declare>
        [ <embedded character set declaration> ]
        [ <host variable definition>... ]
      <embedded SQL end declare>
    | <embedded SQL MUMPS declare>

<embedded SQL begin declare> ::=
    <SQL prefix> BEGIN DECLARE SECTION
        [ <SQL terminator> ]

<SQL prefix> ::=
      EXEC SQL
    | <ampersand>SQL<left paren>

<SQL terminator> ::=
      END-EXEC
    | <semicolon>
    | <right paren>

<embedded character set declaration> ::=
    SQL NAMES ARE <character set specification>

<host variable definition> ::=
      <Ada variable definition>
    | <C variable definition>
    | <COBOL variable definition>
    | <Fortran variable definition>
    | <MUMPS variable definition>
    | <Pascal variable definition>
```

```
    | <PL/I variable definition>

<Ada variable definition> ::=
    <Ada host identifier> [ ( <comma> <Ada host identifier> )... ] :
    <Ada type specification> [ <Ada initial value> ]

<Ada type specification> ::=
      <Ada qualified type specification>
    | <Ada unqualified type specification>

<Ada qualified type specification> ::=
      SQL_STANDARD.CHAR [ CHARACTER SET
         [ IS ] <character set specification> ]
          <left paren> 1 <double period> <length> <right paren>
    | SQL_STANDARD.BIT
          <left paren> 1 <double period> <length> <right paren>
    | SQL_STANDARD.SMALLINT
    | SQL_STANDARD.INT
    | SQL_STANDARD.REAL
    | SQL_STANDARD.DOUBLE_PRECISION
    | SQL_STANDARD.SQLCODE_TYPE
    | SQL_STANDARD.SQLSTATE_TYPE
    | SQL_STANDARD.INDICATOR_TYPE

<Ada unqualified type specification> ::=
      CHAR
          <left paren> 1 <double period> <length> <right paren>
    | BIT
          <left paren> 1 <double period> <length> <right paren>
    | SMALLINT
    | INT
    | REAL
    | DOUBLE_PRECISION
    | SQLCODE_TYPE
    | SQLSTATE_TYPE
    | INDICATOR_TYPE

<Ada initial value> ::=
    <Ada assignment operator> <character representation>...

<Ada assignment operator> ::= <colon><equals operator>

<C variable definition> ::=
      [ <C storage class> ]
      [ <C class modifier> ]
      <C variable specification>
    <semicolon>

<C storage class> ::=
      auto
    | extern
    | static

<C class modifier> ::= const | volatile
```

```
<C variable specification> ::=
      <C numeric variable>
    | <C character variable>
    | <C derived variable>

<C numeric variable> ::=
    ( long | short | float | double )
      <C host identifier> [ <C initial value> ]
            [ ( <comma> <C host identifier> [ <C initial value> ] )... ]

<C initial value> ::=
    <equals operator> <character representation>...

<C character variable> ::=
    char [ CHARACTER SET
             [ IS ] <character set specification> ]
      <C host identifier>
        <C array specification> [ <C initial value> ]
        [ ( <comma> <C host identifier>
          <C array specification>
                 [ <C initial value> ] )... ]

<C array specification> ::=
    <left bracket> <length> <right bracket>

<C derived variable> ::=
      <C VARCHAR variable>
    | <C bit variable>

<C VARCHAR variable> ::=
    VARCHAR [ CHARACTER SET [ IS ]
        <character set specification> ]
        <C host identifier>
            <C array specification> [ <C initial value> ]
          [ ( <comma> <C host identifier>
              <C array specification>
                      [ <C initial value> ] )... ]

<C bit variable> ::=
    BIT <C host identifier>
        <C array specification> [ <C initial value> ]
      [ ( <comma> <C host identifier>
        <C array specification>
                   [ <C initial value> ] )... ]

<COBOL variable definition> ::=
    (01|77) <COBOL host identifier> <COBOL type specification>
      [ <character representation>... ] <period>

<COBOL type specification> ::=
      <COBOL character type>
    | <COBOL bit type>
    | <COBOL numeric type>
    | <COBOL integer type>
```

```
<COBOL character type> ::=
    [ CHARACTER SET [ IS ]
          <character set specification> ]
    ( PIC | PICTURE ) [ IS ] ( X [ <left paren> <length> <right paren> ] )...

<COBOL bit type> ::=
    ( PIC | PICTURE ) [ IS ]
        ( B [ <left paren> <length> <right paren> ] )...

<COBOL numeric type> ::=
    ( PIC | PICTURE ) [ IS ]
      S <COBOL nines specification>
    [ USAGE [ IS ] ] DISPLAY SIGN LEADING SEPARATE

<COBOL nines specification> ::=
      <COBOL nines> [ V [ <COBOL nines> ] ]
    | V <COBOL nines>

<COBOL nines> ::= ( 9 [ <left paren> <length> <right paren> ] )...

<COBOL integer type> ::=
      <COBOL computational integer>
    | <COBOL binary integer>

<COBOL computational integer> ::=
    ( PIC | PICTURE ) [ IS ] S<COBOL nines>
      [ USAGE [ IS ] ] ( COMP | COMPUTATIONAL )

<COBOL binary integer> ::=
    ( PIC | PICTURE ) [ IS ] S<COBOL nines>
      [ USAGE [ IS ] ] BINARY

<Fortran variable definition> ::=
    <Fortran type specification>
    <Fortran host identifier>
        [ ( <comma> <Fortran host identifier> )... ]

<Fortran type specification> ::=
      CHARACTER [ <asterisk> <length> ]
          [ CHARACTER SET [ IS ]
                <character set specification> ]
    | BIT [ <asterisk> <length> ]
    | INTEGER
    | REAL
    | DOUBLE PRECISION

<MUMPS variable definition> ::=
    ( <MUMPS numeric variable> | <MUMPS character variable> )
        <semicolon>

<MUMPS numeric variable> ::=
    <MUMPS type specification>
      <MUMPS host identifier> [ ( <comma> <MUMPS host identifier> )... ]

<MUMPS type specification> ::=
```

```
      INT
    | DEC
          [ <left paren> <precision> [ <comma> <scale> ] <right paren> ]
    | REAL

<MUMPS character variable> ::=
    VARCHAR <MUMPS host identifier> <MUMPS length specification>
      [ ( <comma> <MUMPS host identifier> <MUMPS length specification> )... ]

<MUMPS length specification> ::=
    <left paren> <length> <right paren>

<Pascal variable definition> ::=
    <Pascal host identifier> [ ( <comma> <Pascal host identifier> )... ] <colon>
      <Pascal type specification> <semicolon>

<Pascal type specification> ::=
      PACKED ARRAY
          <left bracket> 1 <double period> <length> <right bracket>
        OF CHAR
          [ CHARACTER SET [ IS ]
                <character set specification> ]
    | PACKED ARRAY
          <left bracket> 1 <double period> <length> <right bracket>
        OF BIT
    | INTEGER
    | REAL
    | CHAR [ CHARACTER SET
                                [ IS ] <character set specification> ]
    | BIT

<PL/I variable definition> ::=
    (DCL | DECLARE)
        (   <PL/I host identifier>
          | <left paren> <PL/I host identifier>
                [ ( <comma> <PL/I host identifier> )... ] <right paren> )
    <PL/I type specification>
    [ <character representation>... ] <semicolon>

<PL/I type specification> ::=
      ( CHAR | CHARACTER ) [ VARYING ]
          <left paren><length><right paren>
          [ CHARACTER SET
                [ IS ] <character set specification> ]
    | BIT [ VARYING ] <left paren><length><right paren>
    | <PL/I type fixed decimal> <left paren> <precision>
          [ <comma> <scale> ] <right paren>
    | <PL/I type fixed binary> [ <left paren> <precision> <right paren> ]
    | <PL/I type float binary> <left paren> <precision> <right paren>

<PL/I type fixed decimal> ::=
      ( DEC | DECIMAL ) FIXED
    | FIXED ( DEC | DECIMAL )

<PL/I type fixed binary> ::=
```

```
      ( BIN | BINARY ) FIXED
    | FIXED ( BIN | BINARY )

<PL/I type float binary> ::=
      ( BIN | BINARY ) FLOAT
    | FLOAT ( BIN | BINARY )

<embedded SQL end declare> ::=
    <SQL prefix> END DECLARE SECTION
        [ <SQL terminator> ]

<embedded SQL MUMPS declare> ::=
    <SQL prefix>
      BEGIN DECLARE SECTION
        [ <embedded character set declaration> ]
        [ <host variable definition>... ]
      END DECLARE SECTION
    <SQL terminator>

<embedded SQL statement> ::=
    <SQL prefix>
      <statement or declaration>
    [ <SQL terminator> ]

<statement or declaration> ::=
      <declare cursor>
    | <dynamic declare cursor>
    | <temporary table declaration>
    | <embedded exception declaration>
    | <SQL procedure statement>

<embedded exception declaration> ::=
    WHENEVER <condition> <condition action>

<condition> ::=
    SQLERROR | NOT FOUND

<condition action> ::=
    CONTINUE | <go to>

<go to> ::=
    ( GOTO | GO TO ) <goto target>

<goto target> ::=
      <host label identifier>
    | <unsigned integer>
    | <host PL/I label variable>

<host label identifier> ::= !!<EMPHASIS>(See the Syntax Rules.)

<host PL/I label variable> ::= !!<EMPHASIS>(See the Syntax Rules.)

<preparable statement> ::=
      <preparable SQL data statement>
    | <preparable SQL schema statement>
```

```
    | <preparable SQL transaction statement>
    | <preparable SQL session statement>
    | <preparable implementation-defined statement>

<preparable SQL data statement> ::=
      <delete statement: searched>
    | <dynamic single row select statement>
    | <insert statement>
    | <dynamic select statement>
    | <update statement: searched>
    | <preparable dynamic delete statement: positioned>
    | <preparable dynamic update statement: positioned>

<dynamic single row select statement> ::= <query specification>

<dynamic select statement> ::= <cursor specification>

<preparable dynamic delete statement: positioned> ::=
   DELETE [ FROM <table name> ]
      WHERE CURRENT OF <cursor name>

<preparable dynamic update statement: positioned> ::=
   UPDATE [ <table name> ]
      SET <set clause list>
      WHERE CURRENT OF <cursor name>

<preparable SQL schema statement> ::=
      <SQL schema statement>

<preparable SQL transaction statement> ::=
      <SQL transaction statement>

<preparable SQL session statement> ::=
      <SQL session statement>

<preparable implementation-defined statement> ::=
    !! <EMPHASIS>(See the Syntax Rules.)

<direct SQL statement> ::=
    <directly executable statement> <semicolon>

<directly executable statement> ::=
      <direct SQL data statement>
    | <SQL schema statement>
    | <SQL transaction statement>
    | <SQL connection statement>
    | <SQL session statement>
    | <direct implementation-defined statement>

<direct SQL data statement> ::=
      <delete statement: searched>
    | <direct select statement: multiple rows>
    | <insert statement>
    | <update statement: searched>
    | <temporary table declaration>
```

```
<direct select statement: multiple rows> ::=
    <query expression> [ <order by clause> ]

<direct implementation-defined statement> ::=
    !!<EMPHASIS>(See the Syntax Rules)

<SQL object identifier> ::=
    <SQL provenance> <SQL variant>

<SQL provenance> ::= <arc1> <arc2> <arc3>

<arc1> ::= iso | 1 | iso <left paren> 1 <right paren>

<arc2> ::= standard | 0 | standard <left paren> 0 <right paren>

<arc3> ::= 9075

<SQL variant> ::= <SQL edition> <SQL conformance>

<SQL edition> ::= <1987> | <1989> | <1992>

<1987> ::= 0 | edition1987 <left paren> 0 <right paren>

<1989> ::= <1989 base> <1989 package>

<1989 base> ::= 1 | edition1989 <left paren> 1 <right paren>

<1989 package> ::= <integrity no> | <integrity yes>

<integrity no> ::= 0 | IntegrityNo <left paren> 0 <right paren>

<integrity yes> ::= 1 | IntegrityYes <left paren> 1 <right paren>

<1992> ::= 2 | edition1992 <left paren> 2 <right paren>

<SQL conformance> ::= <low> | <intermediate> | <high>

<low> ::= 0 | Low <left paren> 0 <right paren>

<intermediate> ::= 1 | Intermediate <left paren> 1 <right paren>

<high> ::= 2 | High <left paren> 2 <right paren>
```

B Παράρτημα B - Οδηγός Εκμάθησης SQL για αρχάριους

B.1 Οδηγός Εκμάθησης για PostgreSQL

Μαζί με τη διανομή της PostgreSQL διανείμεται οδηγός εκμάθησης SQL. Τα scripts του οδηγού εκμάθησης SQL βρίσκονται στον κατάλογο src/tutorial

B.2 Δείκτες Internet διευθύνσεων

Οι αρχάριοι μπορούν να βρουν οδηγό εκμάθησης SQL στο

- <http://w3.one.net/~jhoffman/sqltut.htm>

Σχόλια ή υποδείξεις; Στείλτε mail στο

- jhoffman@one.net

Τα παρακάτω sites προτείνονται από τον John Hoffman:

- Εγχειρίδιο SQL <http://www.contrib.andrew.cmu.edu/~shadow/sql.html>
- Επικοινωνήστε με την SQL Pro <http://www.inquiry.com/techtips/thesqlpro/>
- Χρήσιμα Sites για τη σχεσιακή βάση δεδομένων SQL Pro <http://www.inquiry.com/techtips/thesqlpro/usefulsites.html>
- Κώδικας για προγραμματιστές <http://infoweb.magi.com/~steve/develop.html>
- DBMS Sites <http://info.itu.ch/special/wwwfiles> Πηγαίνετε εδώ και δείτε το αρχείο comp_db.html
- DB Ingredients <http://www.compapp.dcu.ie/databases/f017.html>
- Web Authoring <http://www.stars.com/Tutorial/CGI/>
- Computing Dictionary <http://wfn-shop.princeton.edu/cgi-bin/foldoc>
- DBMS Lab/Links <http://www-ccs.cs.umass.edu/db.html>
- SQL FAQ <http://epoch.CS.Berkeley.EDU:8000/sequoia/dba/montage/FAQ> Πηγαίνετε εδώ και δείτε το αρχείο SQL_TOC.html
- Βάσεις SQL <http://chaos.mur.csu.edu.au/itc125/cgi/sqldb.html>
- RIT Σελίδα για Σχεδιασμό Βάσεων <http://www.it.rit.edu/~wjs/IT/199602/icsa720/icsa720postings.html>
- Database Jump Site <http://www.pcslink.com/~ej/dbweb.html>
- Βοηθήματα προγραμματισμού στο Web <http://www.eng.uc.edu/~jtilley/tutorial.html>
- Πηγές για Προγραμματιστές <http://www.ndev.com/ndc2/support/resources.htp>
- Λίστα ερωτήσεων <http://ashok.pair.com/sql.htm>
- Διάφορα IMAGE SQL <http://jazz.external.hp.com/training/sqltables/main.html>
- Internet Resource List <http://www.eit.com/web/netservices.html>

C Appendix C - Οδηγίες για γρήγορη εγκατάσταση Linux

Αν σκοπεύετε να χρησιμοποιήσετε τη PostgreSQL σε Linux, και χρειάζεστε βοήθεια κατά την εγκατάσταση στο Linux, επισκεφτείτε τους δείκτες που θα βρείτε στο Appendix. Καλύπτουν τα παρακάτω θέματα -

- Χαρακτηριστικά του Linux - Γιατί το Linux είναι καλύτερο για εξυπηρετητή βάσεων δεδομένων σε σύγκριση με τα Windows 95/NT
- Οδηγίες για 10λεπτη γρήγορη εγκατάσταση Linux
- Λίστα αναλογιών Microsoft-Linux
- Γρήγορα βήματα για να μεταγλωττίσετε τον πυρήνα Linux

- Main Site is at `<http://members.spree.com/technology/aldev/>`
- Mirror site `<http://aldev.8m.com>`
- Mirror site `<http://aldev.webjump.com>`
- Mirror site `<http://homepages.infoseek.com/~aldev1/index.html>`
- Mirror site `<http://www3.bcity.com/aldev/>`