



**About Us**  
Innovative Routines International (IRI), Inc.

- Founded 1978, New York
- Released world's 1<sup>st</sup> commercial sorts for: CPM, DOS, UNIX, Windows, iSeries
- 25 HQ Staff + 30 Int'l Agencies
- Profitable since inception
- Owner-run, Single-focus
- Experts in sorting, DB / SQL, DW / ETL
- Recommended by all HW OEMs
- Embedding by/with top ISVs: Acucobol, Ascential, CA, Cincom, Exorcan, Logica, Merse, Soliton, Sabre, Informatica, Kalido, VIPS, etc.

Innovative Routines International, Inc.  
2194 Highway A1A, Suite 303  
Melbourne, FL 32937 432 USA  
www.cosort.com  
1-800-333-SORT / T-321-777-8889  
FAX: 1-321-777-8889

**Our Users' Issues**

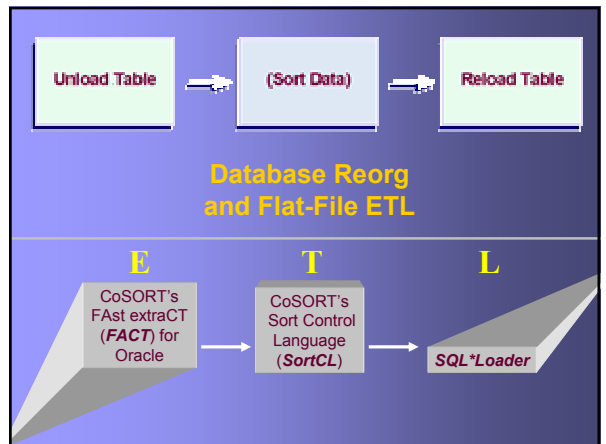
- **Database > 1 million rows**
  - Slow query responses
  - Long unload and/or load times
  - Poor RDBMS or hardware performance
- **Growing a data warehouse/webhouse**
  - Need fast integration and staging of data
  - ETL tools sort, join, and aggregate slowly
  - Hand-off to data marts, BI, and CRM tools
- **The current sort is not CoSORT**
  - Need to sort faster, do more, pay less



**Flat-File ETL Operations**

**CoSORT's SortCL**  
is used for very fast data integration and staging; extract – transform – load (ETL) operations on multiple, high-volume external, (sequential) data sources.

**TDWI's Customer Intelligence Lifecycle**



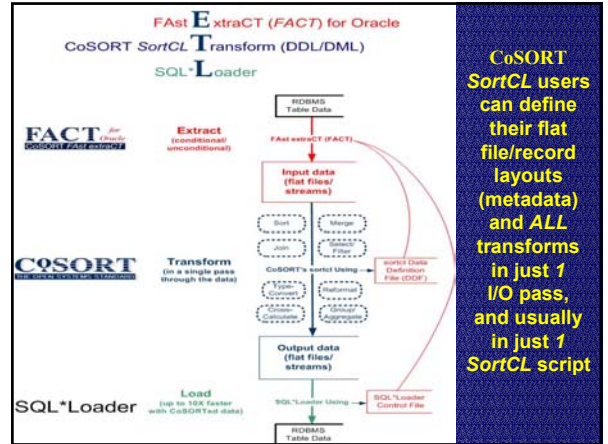
## Using CoSORT to Perform and/or Accelerate Data Warehouse ETL

### Use CoSORT's *FACT* & *SortCL* for Flat-File ETL Tool

- Row/column extraction, cleansing, aggregation, integration, ordering, converting, segmenting, reformatting, and load acceleration
- ETL operations can be performed by CoSORT much faster on flat files, and usually in a single script, and in a single pass through the data

### Use CoSORT Tools to Speed Third-Party ETL Tools

- Informatica PowerCenter**
  - Exclusive CoSORT AEP sorts 10X faster than nSorter Tx
  - Or, use CoSORT/SortCL pre-session sort and cleanse. Aggregate with 'sorted ports' option.
- Ascential DataStage XE**
  - Exclusive CoSORT Plug-In sorts 10x faster than native stage
  - Or, add Sequential File Stage prior to aggregation using CoSORTed input and output files. Define sorted fields to aggregation stage.
- Sagent, Kalido, Embarcadero, Ab Initio**
  - Existing or pending direct hooks to CoSORT or SortCL engine(s)



CoSORT *SortCL* users can define their flat file/record layouts (metadata) and ALL transforms in just 1 I/O pass, and usually in just 1 *SortCL* script

## Faster Database Unloads

- Remove DISTINCT, WHERE, GROUP BY, ORDER BY from SELECT statement ... it's a simpler SQL statement
- Unencumbered unloads are also up to 40% faster
- Even faster is CoSORT's *Fast extraCT (FACT)* for Oracle product which also builds the unload's metadata for *SortCL* and *SQL\*Loader* for same-job transforms and loads!

### External transform are faster ...

- Filter, sort, join, summarize and reformat the 'dumped' flat files with CoSORT's *SortCL* after unloading

### Additional Benefits:

- Processing flat files is faster than structured DB tables
- CoSORTing is more efficient than internal DB/SQL sorting
- The DB does not have to store temporary sort files

**ETL**

CoSORT's *Fast extraCT (FACT)* for Oracle

*New Product!*

## Faster Oracle Unloads

- High-speed, parallel unloads (Extract)**
  - Conditional or unconditional (faster) SQL table dumps to flat-files
- Creates CoSORT/SortCL Metadata (Transform)**
  - Simultaneously select, join, sort, aggregate, calculate, convert, re-format
- Creates SQL\*Loader Metadata (Load)**
  - Pre-sorted flat files repopulate Oracle up to 90% faster

**Combine all 3 into 1 Fast E-T-L Pass! See demo ...**

Fast data integration & staging (transform) all-in-one I/O pass ...

**ETL**

CoSORT's Sort Control Language (**SortCL**)

... through many, large, differently-formatted inputs:

- Select** via record filters or conditional include/omit
- Sort / Merge** on any number of keys in any position
- Join** SQL match syntax 1-1, many-1, inner & outer
- Convert** translate input field data types to new types
- Aggregate** min, max, average, sum, count (sub and grand)
- Calculate** across rows to perform math (+ sci functions)
- Re-map** change field positions, sizes, and values
- Report** to highly-formatted, multi-level output targets
- User Exits** for custom input, compare and output criteria

## Faster Database Reloads

- Index creation = sorting + writing index
- CoSORTing is more efficient than internal DB (SQL) sorting
- Pre-sorting is recommended by Oracle, Sybase, Microsoft SQL Server and IBM DB2 (as well as other ETL ISVs)
- External (flat-file) pre-CoSORTs speed loads up to 90%
- Having many indexed tables will speed more queries
- SQL\*Loader Control File metadata already there from *FACT*!
- CoSORT Load Accelerator for DB2 (CLA4DB2, co-invented with IBM in 1999) directly interfaces with DB2's load utility and cuts load times up to 50%

**CoSORT**  
THE OPEN SYSTEMS STANDARD

## Faster Oracle Reloads

SQL\*Loader **ETL**

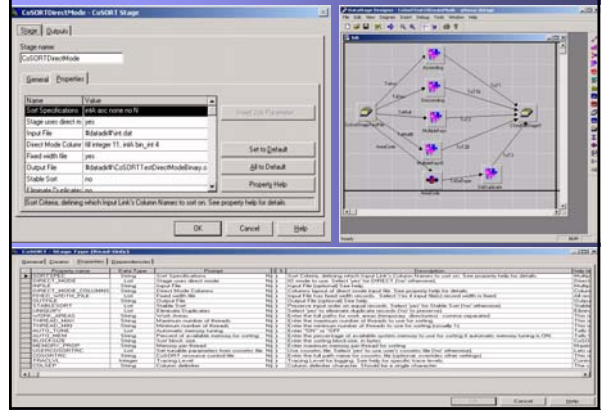
- 1) Use SQL SELECT WHERE or CoSORT's new *FAst extraCT (FACT)* for Oracle to **dump** the table data to (a) flat file(s);
- 2) Pre-sort the input on the longest index field, using any of CoSORT's end-user utilities (e.g. *SortCL*) or APIs;
- 3) Use SQL\*LOADER to **load** the sorted file(s), with the command line argument **DIRECT=TRUE**; and,
- 4) To **create indexes**:

**during** the load, specify **SORTED INDEXES** in the SQL\*Loader Control file (.ctl)  
**after** the load, use the SQL CREATE INDEX command with the NOSORT option.

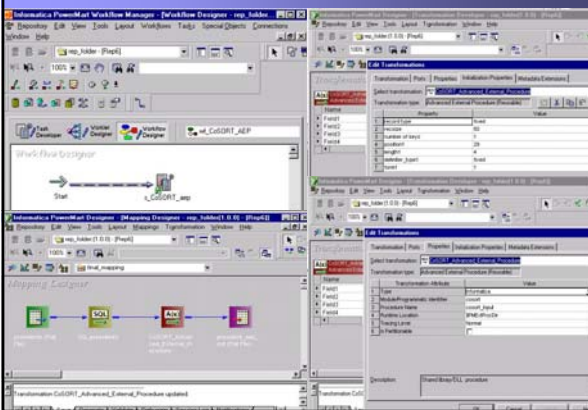
For more details on pre-sorting before loading, see the Oracle Server Utilities User's Guide:  
 "You can improve the performance of direct path loads by pre-sorting your data on indexed columns. Pre-sorting minimizes the temporary storage requirements during the load, and allows you to take advantage of high-performance sorting routines that are optimized for your operating system and application."



## CoSORT Plug-In for Ascential's DataStage XE



## CoSORT Advanced External Procedure for Informatica

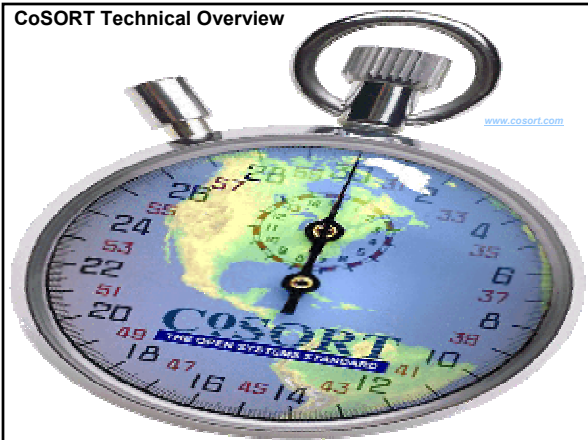


### ~ 15 Million Row Benchmarks

BMC Oracle Unload	25 minutes
<b>CoSORT FACT</b>	10 minutes
	<i>2.5 times faster</i>
MF COBOL Sort	25 minutes
<b>MF CoSORT</b>	5 minutes
	<i>3 times faster</i>
Ascential DataStage	20 minutes
Informatica Nsort Tx	20 minutes
<b>CoSORT Plug-In Sort</b>	2 minutes
	<i>~10 times faster</i>
Oracle Express Load	9 hours
<b>CoSORTed Load</b>	41 minutes
	<i>~10 times faster</i>
Oracle PL/SQL Join	1 hour
<b>CoSORT Sort/Join</b>	3 minutes
	<i>20 times faster</i>

*It's about time...*

## CoSORT Technical Overview



## CoSORT Functionality

- **High-Performance Record Processing**
  - Sort, Copy, Merge, Join, Filter, Reformat, Aggregate, Summarize, Data Partitioning, Data conversion, Cross-Calculation, Output Sequence, Custom Reporting
- **Integrates Multiple Data Sources**
  - Sequential files
  - Unisys VB files
  - Named & Unnamed Pipes
  - Table(s) of records in memory
  - Individual records in memory
  - Custom input and output procedures (User Exits)
  - Developing link to DataFlux (Data Quality/Cleansing)

## CoSORT Technical Specifications

### Supporting 100+ data types

Binary integer	Edited numeric
Bit	Embedded sign
Character	Float
Date/time	Packed decimal
Double	Unsigned decimal

### Multiple Collating Sequences

- Standard (ASCII, Folded ASCII, EBCDIC, MF and RM COBOL, Multinational, Timestamps)
- ACUCOBOL-GT Vision (index) files
- LOCALE (operating system defined), IP Addresses
- User-defined (multiple compare procedures)
- 'mcs' API supports 1-pass, multi-sort jobs



**Sort  
Benchmarks**

## Coroutine Architecture, SMP Performance

- 19 MB in 2 seconds on P200/2 w/ NT, 2 keys
- 1.8 GB in 67 seconds on Compaq GS140/8
- 2.4 GB in 39 seconds on SunFire 15K / 6CPUs
- 5.2 GB in 20 minutes on Sun UE3000, 23 keys
- 30 GB in 45 minutes on Linux 4-CPU Itanium1
- 272 GB in <2 hours on IBM Numa-Q 2000/64  
*setting a 2000 TPC-H DSS benchmark record.*

**CoSORT**  
THE OPEN SYSTEMS STANDARD

## CoSORT User Interfaces



- **Sort Replacements**
  - Drop-in (plug 'n play) for many third-party sorts
- **Sort-I (Sort Interactive)**
  - Command line prompt/batch program for novices
- **SortCL (Sort Control Language)**
  - high-level DDL/DML program for data warehouse data integration and staging (ETL), and reporting
  - Java GUI for cross-platform job design/launch
- **SortCL Conversion Tools**
  - for third-party metadata, mainframe sort parms
- **Application Programming Interfaces (APIs)**
  - 3 callable libraries for third-party software integration

## Seamless Migrations



### Plug-In Sort Replacements for:

- Amdocs Ensemble
- Ascential DataStage
- Cincom Supra SQL
- IBM DB2 UDB Loader
- Informatica PowerMart/Center
- MF COBOL
- SAS System
- Software AG Natural
- Sun Mainframe Rehosting's MTP/MBM
- SVR4 Unix /bin
- SyncSort UNIX

### and Metadata Converters for:

- MVS and VSE JCL sort parms
- DEC VAX VMS sorts
- COBOL FD metadata
- Common & Extended Web Log Format
- Microsort CSV files

**CoSORT**  
THE OPEN SYSTEMS STANDARD

Fast data integration  
& staging (transformations)  
all-in-one I/O pass ...

ETL

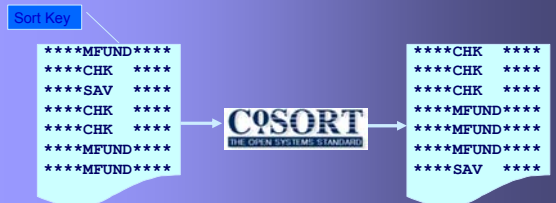
CoSORT's  
Sort Control  
Language  
(SortCL)

... through many,  
large, differently-  
formatted inputs:

- **Select** via record filters or conditional include/omit
- **Sort / Merge** on any number of keys in any position
- **Join** SQL match syntax 1-1, many-1, inner & outer
- **Convert** translate input field data types to new types
- **Aggregate** min, max, average, sum, count (sub and grand)
- **Calculate** across rows to perform math (+ sci functions)
- **Re-map** change field positions, sizes, and values
- **Report** to highly-formatted, multi-level output targets
- **User Exits** for custom input, compare and output criteria

## SortCL Data Processing Functions Sorting

Order data according to one or more  
character, numeric or date key fields.





## SortCL Data Processing Functions Match / Join

Join fields from records in one file to records in another file according to one or more key fields. SortCL uses SQL syntax for inner, left, right and full outer joins. Join users can also remove matches from outer joins, and aggregate and/or cross-calculate the joined results in the same report.

Doe, Jane, 65 Black St., 32901      Manhasset, NY, 11030  
Smith, John, 12 Apple St., 11030      Melbourne, FL, 32901



Doe, Jane, 65 Black St., Melbourne, FL, 32901  
Smith, John, 12 Apple St., Manhasset, NY, 11030

## SortCL Data Processing Functions Select / Filter

- Conditionally retain or eliminate records
- Conditionally or unconditionally retain or eliminate fields within the record
- Conditions can be elaborate (IF-THEN-ELSE, ELSE IF)

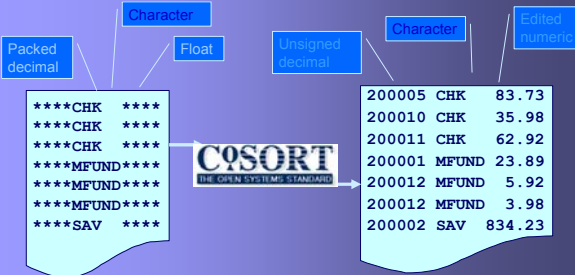
```
200005 CHK 83.73
200010 CHK 35.98
200011 CHK 62.92
200001 MFUND 23.89
200012 MFUND 5.92
200012 MFUND 3.98
200002 SAV 834.23
```



```
CHK, 83.73
CHK, 35.98
CHK, 62.92
SAV, 834.23
```

## SortCL Data Processing Functions Conversion

Universal Data Type Translation.



## SortCL Data Processing Functions Aggregation

Summarize records to eliminate duplicates, and build structured reports with static and/or running sub and grand totals, min, max, average, and count fields.

```
CHK, 83.73
CHK, 35.98
CHK, 62.92
SAV, 834.23
```



```
CHK, 182.63
SAV, 834.23
```

## SortCL Data Processing Functions Data Partitioning

Conditionally segment one or more source files to different target outputs.



## SortCL Data Processing Functions Output Sequencing

- SortCL can add a unique row identifier number to output
- Numbering can start at any value – negative, 0, or positive
- Useful for creating indexes in databases

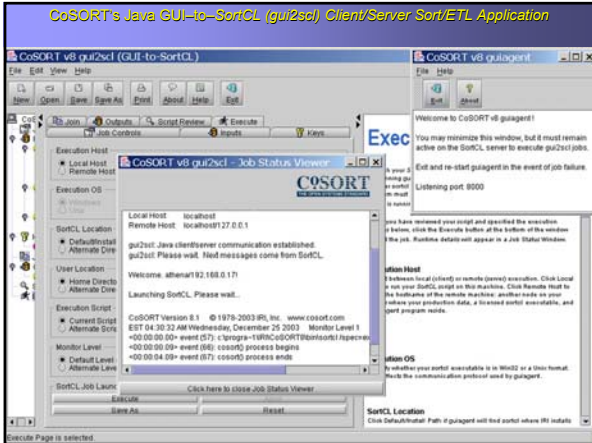
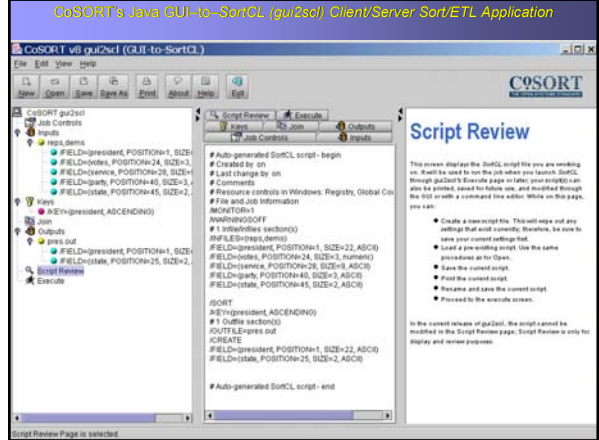
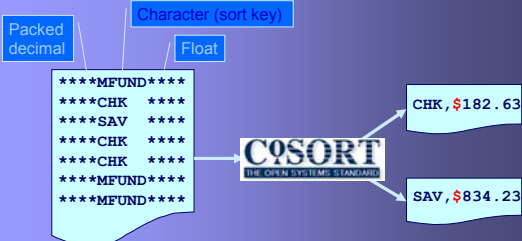
```
HJ28983, shirt, red, 12.49, 2
HJ28983, shirt, blue, 12.49, 1
IL93848, pants, blue, 32.11, 3
```



```
0001, IL93848, pants, blue, 32.11, 3
0002, HJ28983, shirt, blue, 12.49, 1
0003, HJ28983, shirt, red, 12.49, 2
```

# SortCL Data Processing Functions 1-Pass Data Staging

Sorted, joined, converted, filtered, aggregated, reformatted, and partitioned, all in one I/O pass.



## SortCL Benefit Summary

- Data staging operations are a combination of ordering, transforming, joining and summarizing – all of which SortCL easily performs.
- Eliminate processing bottlenecks; stay within the staging window.
- Processing flat files with CoSORT is the fastest possible means.
- Java GUI reduces development time and improves maintainability.
- Programming or DBA expertise not required. SortCL is easier to code and runs faster than PL/SQL, PERL, C and COBOL programs.
- Explicit, completed application looks like the original specification.
- Higher level of abstraction simplifies programs defined with views, such as records and fields, versus buffers and bytes.
- SortCL layout files separate data and metadata from applications.
- CoSORT/SortCL syntax is familiar to mainframers and C/UNIX DBAs.

## Now You Know.

CoSORT is a world leader in UNIX and Windows sorting, and a critical infrastructure tool for the staging, integrating, manipulating and presenting of high data volumes. Since 1978, leading IT installations have chosen CoSORT to satisfy their project requirements and performance needs in:

- Mainframe JCL and COBOL sort migrations
- Database reorg (unload, sort, reorg)
- Data warehouse staging (ETL)
- Detail and summary reporting
- Third-party sort conversion / replacement
- New batch processes and software products

Innovative Business International, Inc.  
2194 Highway A1A, Suite 303  
Melbourne, FL 32937 432 USA  
www.cosort.com  
1-800-833-SORT (1-326-777-8888)  
Fax: 407-567-7888