

Converge to Silicon Success

Milkyway and Bridging to OA

**Laurence Brevard
Strategic Alliances Manager**

**Electronic Design Process Workshop
Monterey, California
April 8, 2005**

SYNOPSYS

Contents

- **Milkyway history and overview**
- **MAP-in Program**
- **Golden Gate Working Group**
- **Milkyway to OA translator – mw2oa**
 - **Overview**
 - **Mappings**
 - **Serialization**
 - **Other issues**
- **Conclusion**

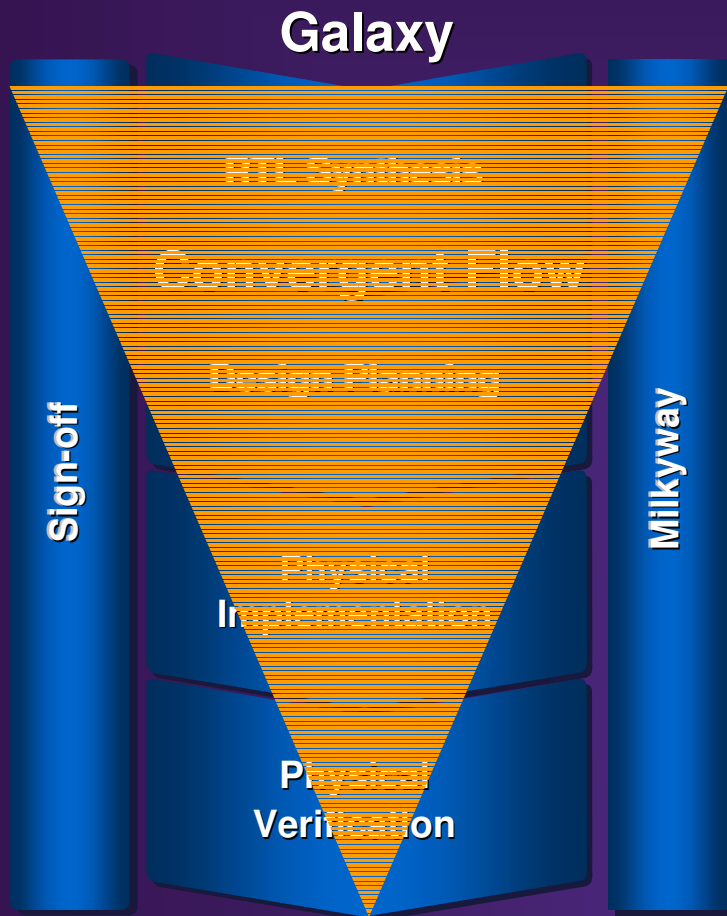
Milkyway Database History

- Used since 1998
- Physical Design and Netlist
- Used by many physical design tools:
 - Apollo, Astro
 - Enterprise, Cosmos, Hercules
 - Jupiter, Star-RCXT
 - Galaxy platform and *new* IC Compiler
- Proven in thousands of tape-outs – now including 65 nm chips
- Now used by MAP-in participants

What is Milkyway?

- **Persistent Database – on disk**
(design data storage system)
- **Objects and operations**
 - available to tools developed by Synopsys
 - seen and manipulated via Scheme and TCL
 - seen and manipulated via C-API
- **Environment for tool and utility operations**
- **The heart of the "Galaxy" Design Platform.**
- **The basis of the Galaxy IC Compiler**

Galaxy 2004 Platform

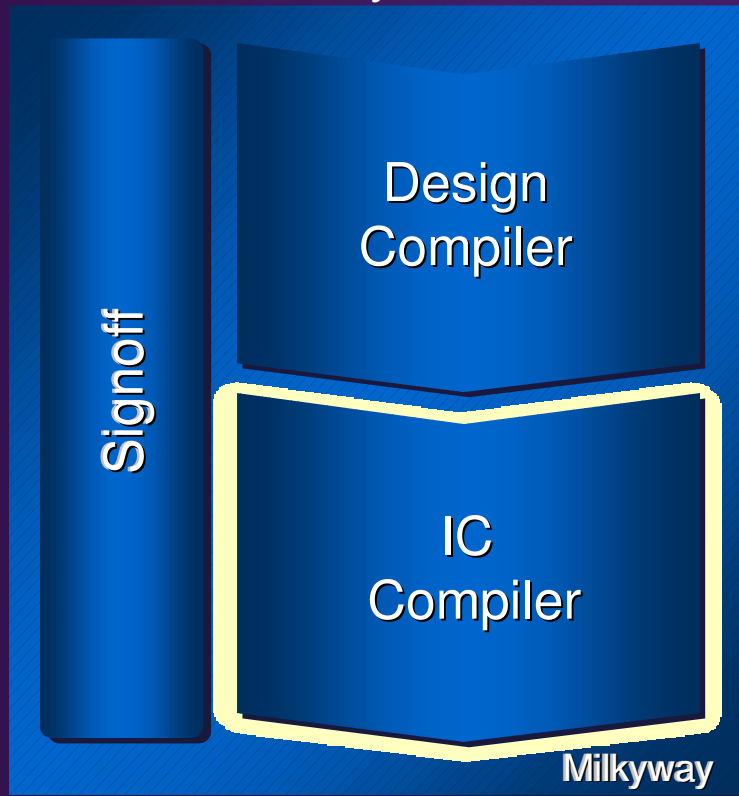


- Industry-leading synthesis
- Patented linear-placement technology
- Yield conformant routing
- Common timing, extraction engines
- Convergent design closure

Introducing IC Compiler

Next-Generation Physical Design System

Galaxy 2005

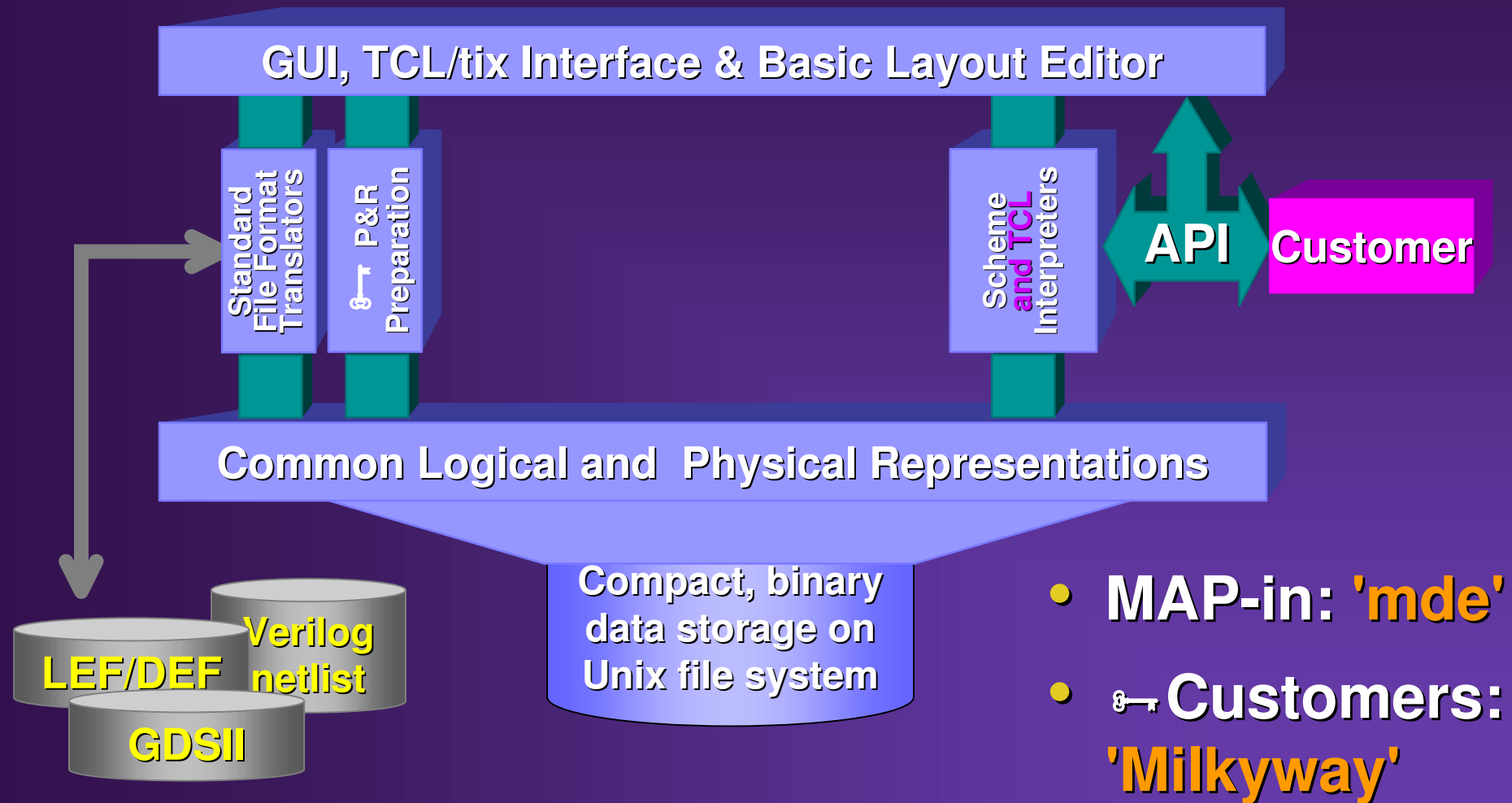


- **Centerpiece of Galaxy 2005**
 - Systemic solution
- **Faster design closure**
 - Extended Physical Synthesis
 - Signoff driven
 - Yield optimization

MAP-in Program for EDA vendors

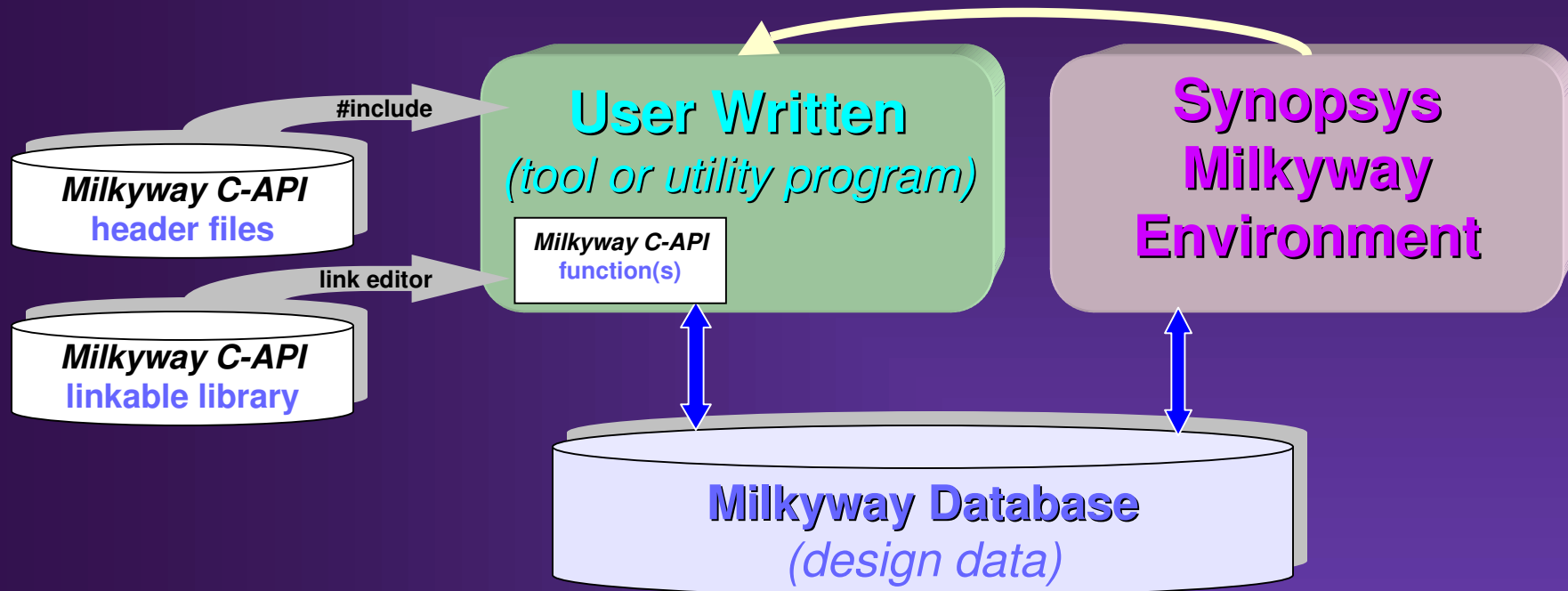
- **Basic Membership Requirements**
 - Commercial software tool vendor that needs access to Milkyway-based data
 - Agree to MAP-in license
- **No Membership Fee**
 - 1st copy of Milkyway Database Environment (MDE) and Milkyway Database Access C-API **at no charge.**
 - Additional copies of software available for modest annual fee
 - Access to SURF on available/approved basis
 - MAP-in web forum for community support
- **Synopsys offers optional support contract**

Milkyway Database Environment



C-API stand-alone Usage Model

If appropriate, the user written program can be called from the Milkyway environment using Scheme (system ...) command



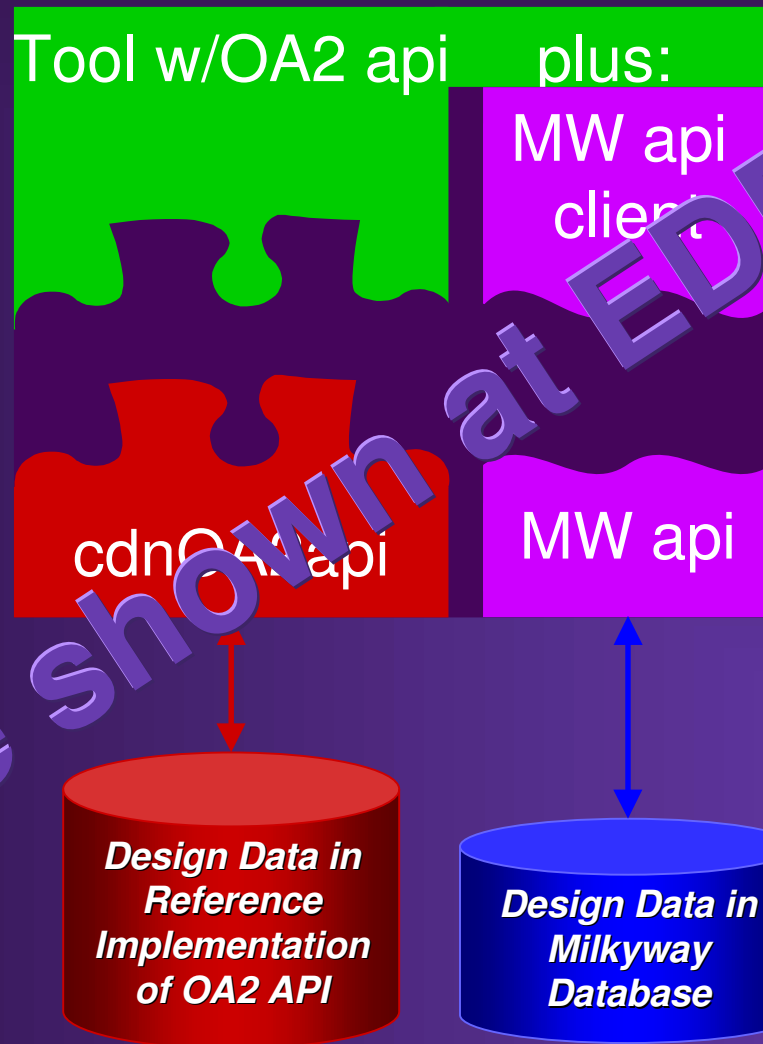
Milkyway evolution...

- **Integration of pre-merger tools from Avant! and Synopsys has gone well.**
- **TCL scripting now in MDE**
- **Better common file format translators with Milkyway as source / destination**
 - **Can build a MW DB using only LEF/DEF**
- **Continuing improvement in the infrastructure for capacity and speed**
- **Future bridges to and from OpenAccess**

Synopsys on GoldenGate Committee

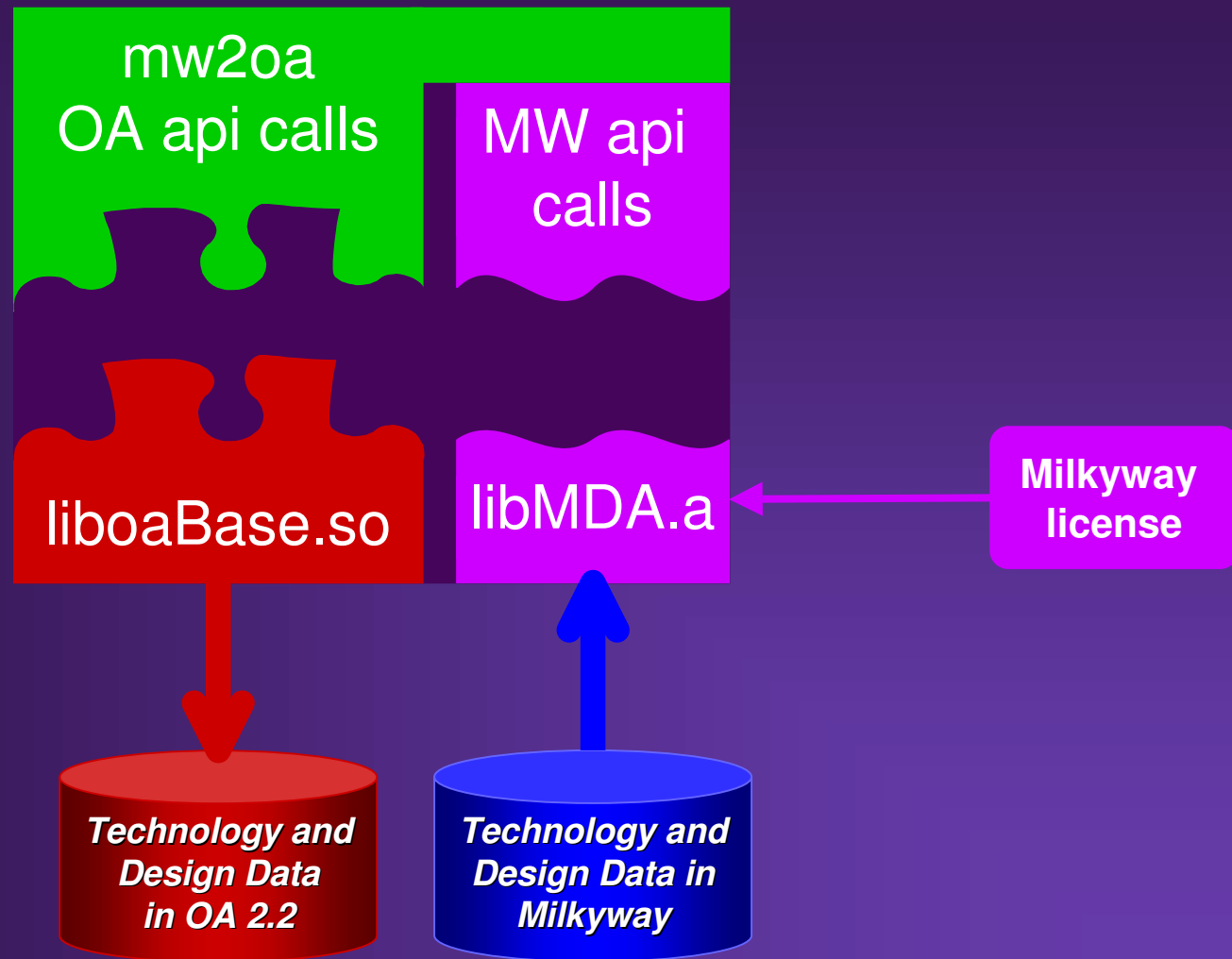
- **OAC formed GoldenGate committee (2003)**
 - **To bridge between Milkyway and OpenAccess.**
- **Synopsys actively supports GoldenGate efforts**
 - *I was the representative from Synopsys in 2003!*
- **CDN committed to build a Milkyway-to-OA bridge**
 - **Initial bridge delivered with OA 2.2 in fall, 2004**
 - *I was the initial programmer!*
- **Synopsys will support an OA-to-Milkyway bridge**
 - **When there is customer data in OA**
 - **When it makes business sense**

OpenAccess and Milkyway APIs



Slide shown at EDP in 2003

A Milkyway to OA bridge



oaMilkyway::mw2oa translator

- Written in 2004 at Cadence
- Uses OpenAccess 2.2 API
- Follows OpenAccess coding standards
- Uses infrastructure common to other OA translators
 - Utility class, Application class, Test infrastructure
- Cadence MAP-in membership provides access to Milkyway C-API, headers, linkable library, license
- Built on 2003.09 MAP-in, initial testing done on 2004.06
- Beta quality
- Initially available to OAC members who are also licensed to use the Milkyway C-API
 - Customers who purchased the appropriate products
 - EDA vendors who are MAP-in members

mw2oa mappings

- Golden Gate working group spreadsheet

Simplified Objects Table						Golden Gate Working Group		
Milkyway						Open A		
Object Type Name	Implementation Notes	Open / Close Operations	Access / Traversal	Creation	Deletion	Object Type Name	Implementation Notes	Open / Close Operations
Technology File	Text file imported into library's "lib" file	Import / Export only	C-API access to some values	Export from "lib" and text editor	Wholesale replace in library. File may be deleted from file system.	Tech. Library	Technology db associated with one or more libraries in search path	Can be done explicitly but is automatic when referring to tech data from cellview data.
Reference Library	Only one library mechanism: file system directory plus "lib" file	Open R only. Any number openable as needed by Design Library	Referenced by name, mappable by Ref. Lib. Ctl. file	Create Library functions. Data preparation steps required to make usable for reference?	Remove all references to this from other libraries. rm -rf ???	Ref. Library	Any library may be a reference library.	RAW access controlled at filesystem level. Possible though not common to write to reflibs.
Design Library	Only one library mechanism: file system directory plus "lib" file	Open R or W. Open Library sets context for Cell operations. Library ObjectID is like a file handle.	Find in file system. Get current library from Cell.	Create Library functions	rm -rf ???	Design Library	Any library may be a design library. You can have multiple design libraries at the same time.	RAW access controlled at filesystem level. Possible and common to write to design libraries.
* Cell	File located under directory corresponding to View type	Open R or W, Save or discard changes, Close, ObjectID is a file handle. Cells may be versioned	Cells in library; see file system db-foreach New C-API iterator	File import, TDF (Scheme), Scheme or C-API	Scheme or C-API	CellView	File set in view directory inside cell directory inside lib directory.	Open R, W, or A, Save or discard changes, Close, ObjectID is an opaque handle. Cells may be versioned using GDM interface and external (not OA-supplied) DM system.

- Next level of mapping came during implementation of the translator
- See April 2004 OA Conference presentation: <http://www.si2.org/oaconf2004/20040405-OAC2004-brevard-mw2oaStatus.pdf>

mw2oa mappings – first order objects

- **TECHNOLOGY**

(via C-API only – not parsing MW Tech File)

- **DBUperUU** → **DBUperUU**
- **contact code** → **oaStdViaDef**
- **layer** → **oaPhysicalLayer**
name and number

- **HIERARCHY**

- **library** → **oaLib**
- **cell** → **oaDesign, oaBlock**

mw2oa mappings – first order – *cont'd*

- **NETLIST**

- **port** → **oaScalarTerm**
- **net** → **oaScalarNet**
- **cellinst** → **oaScalarInst**
- **cellinstMaster** → **oaInstHeader**
- **portinst** → **oaInstTerm**
- **portinstMaster** → private object created automatically

mw2oa mappings – first order – *cont'd*

- **PHYSICAL and TEXT**
 - **pin** → **oaPin**
 - **rectangle** → **oaRect**
 - **polygon** → **oaPolygon, oaPoints**
 - **path** → **oaPath, oaPoints**
 - **boundary** → **oaAreaBoundary or oaPRBoundary**
 - **text object** → **oaText**

mw2oa mappings – first order – *cont'd*

- **ROUTING**

- **contact** → **oaStdVia**
- **wire master** → **oaSegStyle**
- **horizontal or vertical wire** → **oaPathSeg**
- **wire track** → **oaTrackPattern**

- **EXTENSIONS**

- **property on cell or cell object** → **oaProp**
- **group** → **oaGroup**

mw2oa mappings – attributes e.g. 1

- **CellInst** → **oaScalarInst**
 - **owning cell** → **oaBlock**
 - **inst name** → **oaScalarName, oaLefNS**
 - **master name** → **oaScalarName, oaLefNS**
 - **master view** → **oaScalarName, oaLefNS**
 - **position and transform** → **oaTransform**
 - **cellinst master** → **oaInstHeader**

mw2oa mappings – attributes e.g. 2

- **Text Object → oaText**
 - **text string → oaString**
 - **justification → oaTextAlign**
 - **height → oaDist**
 - **position → oaPoint**
 - **rotation, mirror → oaOrient**
 - **font → oaFont**
 - **layer → oaLayerNum**

mw2oa mapping functions

- Virtual functions used for attribute mapping
 - mapTransformCode → oaOrient
 - mapNetType → oaSigType
 - mapPortDirection → oaTermType
 - mapFontCode → oaFont
 - mapPathType → oaPathStyle
 - mapEndStyle → oaEndStyle
 - mapName → oaString

mw2oa conversion granularity

- **Granularity**
 - **One library**
 - **One or more cells of**
 - **One or more views**
- **Convert Library**
 - **Convert technology**
 - **Convert cells and views of interest**

mw2oa serialization – simplifications

- **MWXDb_Get_Objects_ByType**
 - **Used to get all the objects of a given type from a Milkyway cell**
 - **Simplified traversal – e.g., all port instances directly accessible**
- **Ability to create unbound oaScalarInst**
 - **Therefore it not necessary to have a cell definition before creation of an instance that will be bound to it.**

mw2oa serialization – within a cell

- For each cell / view...
 - Instances
 - Nets
 - Ports
 - Port Instances
 - Pins
 - Rectangles
 - Polygons
 - Texts
 - Boundaries
 - Contacts
 - Paths
 - Wire Masters
 - Wires
 - Wire Tracks
 - Cell Properties
 - Groups

mw2oa – serialization improvement

- **Move calls to convert port instances to within the convert instance function**
- **Move calls to convert pins to within convert port function**
- **Remember wire masters after conversion so that oaSegStyle need not be created redundantly**

mw2oa – finding mapped objects

- **Sometimes need to find a corresponding mapped object later**
 - **e.g. connect a port to a net already translated**
- **Used OpenAccess find (by name) where possible**
- **Built hash table of Milkyway object ID to oaObject for objects not findable by name later in OpenAccess**
 - **portinst, rectangle, polygon, text, path**
 - **boundary, contact, wiremaster, wire, group**

Limitations / Issues

- **Layer information such as color and pattern**
 - Stored in Milkyway but in oaPhysicalLayer
 - Could use OA extensions
- **MAP-in C-API missing functions**

NOTE: Some were added in 2004.06 release

 - Getting attached file names / pseudo-names
 - Getting group name
 - Manipulating properties on libraries
 - No ability to set associated net when creating physical objects such as rectangle
(For this translator is only an issue creating test data.)

Limitations / Issues – cont'd

- **Translating routes**
 - **Some rules are available from the Milkyway technology access C-API functions**
 - **Route type codes are not well documented**
 - **No oaRoute creation yet**

Conclusions

- “I’ve looked at love from both sides now!”



- Someone with more tool knowledge could enhance the mappings substantially.
- A mapping layer API is a different project
 - Would be purely incremental
 - Could take clues from the mw2oa program
- **THANK YOU!**