

# Milkyway Database System and MAP-in R&D Update

October 17, 2003

James Hsueh  
Sr. R&D Manager



October 17, 2003

 *Your Interoperability Partner*

## The Story of Milky Way

- Long ago and far away ...
  - Humble beginning with P&R in mind
  - Flexible to accommodate more tools
    - Container and Object
  - Consistent data interpretation
    - Schema, property and attached file
  - The evolution continues ...
    - To bring front and back end applications together, if all possible
- Openness

  *Your Interoperability Partner*

## The Milkyway Access Program

- Not perfect but still improving
  - to be open within the constraint
    - P&R tie (What command available?)
    - GUI environment tie (What API available?)
- The Milkyway Database Environment – mde
  - Access to lots of built-in capabilities
  - Same code base as Astro product releases
- The Milkyway Database Access APIs – libmda.a
  - Enable your application to run directly off Milkyway database



## Changes in The Coming Release

- Baseline 2003.06 MAP-in Release
  - Code base on Milkyway 2003.03
- Changes in 2003.09 MAP-in mde
  - Stroke binding of commands
  - GDS Data Type Support in GUI
  - New verilog reader and data preparation
  - PDEF license change
- New linkable API library (libmda.a )
  - API Basic Concepts
    - Library, Cell, View, Object (netlist and physical)
  - Data access only

# API Basic (if you need this back ground info)

## ■ MWX API

- Predefined set of functions allow user to access Milkyway data by a C or C++ program
- The simple example
  - MWXDb\_Open\_Lib() /\* open a library \*/
  - MWXDb\_Open\_Cell() /\* open a cell \*/
  - MWXDb\_Get\_CellInstance\_ByName()
- Compile and link with libmda.a
  - The program can work on Milkyway database

## ■ Linkable vs. dynamic load library

## API Basic (Cont.)

- What's inside the view
  - CEL -- connectivity and physical
  - FRAM – abstraction of physical library
  - NETL -- netlist
  - TIM – timing models
- What object is accessible
  - Cell instance, port, net
  - Pin, wire, via, path
- What is not available
  - Place\_and\_route\_design
  - Report\_timing

## Technology In Development

- Capacity
  - The Linux challenge – much faster but less memory
  - The memory consumption by Milkyway itself
    - Handle larger design in 32 bit platform
    - More efficient cache management
  - Interactive application vs. batch oriented application
    - Limit Milkyway's memory usage
- Performance is critical too
- Size Matters
  - 32 bit object id is the limitation
    - 10 M instance flat design
    - Virtually no limit for hierarchical design

## Technology Development (cont.)

- Galaxy Platform
  - New requirements from other SNPS tools
    - Physical Compiler, Design Planner
    - Hierarchy manipulation
    - 90nm, 65nm design
  - New and improved utilities
    - Ease-Of-Use in data preparation
    - External file (format) interface
    - SNPS quality process and requirement
  - 3<sup>rd</sup> party compatibility will be maintained
- Will benefit all Milkyway based products
  - Enhancement done in Milkyway will be seen in your app. too

## Coming Soon (New features)

- In mde
  - LM View and Logic View
    - Logic Model and Logic design
  - Plib can be used in place of tf file
    - Liberty and TAP-in
  - LEF/DEF 5.5 support
    - Easier design migration
    - Simplified technology data transfer
- In libmda.a
  - Enhancement for Data Management
  - Memory and performance

## What's Next

- Synchronized with Production Releases
  - Milkyway (mde)
  - C-API (libmda.a)
- Matching scheme/tcl and C-API
  - For database access
  - Tcl command language
- In the Forum
  - Tell us what you like, don't like or would like to see in C-API

# MAP-in Supported Platform

Release	2003.06	2003.09	2004.03/6
Sun 32/64 (Solaris 8)	Y	Y	Y
HP 32 (HP-UX 11.0)	N	Y	Y
HP 64 (HP-UX 11.0)	Y	Y	Y
Linux 32 (RedHat 7.2)	Y	Y	Y
Linux 32 (RHEL v3.0)	N	N	Y
IPF 64 (Linux 7.2)	N	Y	N
IPF 64 (RHEL 2.1)	N	Y	Y
Opteron (RHEL v3.0)	N	Y (1)	Y

(1) – Date TBD

 - Primary (ECS)

 - Secondary (ECS+45 days)

 Your Interoperability Partner

## The libmda.a Beta program

- Various vendors with much different tools in mind
  - Data management
  - Layout editor
- General Feedbacks
  - How can I do such and such tasks ...
  - New API created based on the feedback
    - Deal with wire and wire master
    - library and cell manipulation

## Testcase Available For MAP-in

- Full Adder Testcase (Hierarchical)
  - 2 Half adder & OR Gate
  - Half adder contains XOR & AND Gate
- Top Design Testcase (Hierarchical)
  - 41 Standard Cells
  - 4 Hard Macros
- BM2 Astro Testcase (Flat)
  - 46093 Standard Cells
  - 8 Hard Macros
- ftp download 117M byte

## Where to Get Help

- MAP-in Forum
- Paid Support
- Beta program (or partnership)
- Issues in the tracking
  - Currently there are 9 issues reported in Beta program
    - Some for mde and some for libmda.a
  - More than half issues raised in Beta have been resolved.
- R&D committed to make MAP-in successful
  - Your success is important!

 *Your Interoperability Partner*