

Caterpillar Hydraulic Cylinder Design Environment

Design For 6 Sigma

D. Jason McIntyre
Mahmoud Alrefai
Kalpesh Patel

CATERPILLAR[®]

PRODUCT DEVELOPMENT CONFERENCE



Abstract



CATERPILLAR[®]

In 2001, Caterpillar's Hydraulic and Hydraulic Systems division was under pressure to provide more cylinder designs than ever before, and aggressive goals were being set to reduce our warranty risk. Caterpillar Inc. contacted MSC.Software to determine how we could automate the FEA process, which would allow us to design faster and more robust cylinders. During those planning sessions we laid out a plan to implement a 4 phased project that would not only automate FEA but would standardize the entire hydraulic cylinder design process, from customer request through FEA report generation.

The phase 4 project has allowed Caterpillar to complete 100% FEA's on all components and design calculations, while not increasing the design time from the previous design method. It also captures the design into the database for future access. This allows us to use components from that previous design without having to run a full analysis all over.



6 Sigma DMAIC Methodology

CATERPILLAR®

**Project Broken
Down into several
6 Sigma DMAIC
Projects**

D - Define

M - Measure

A - Analyze

I - Improve

C - Control

Project Completed in 4 Phases

Phase 1 – RDA (Requirements Definition Assessment)

Phase 2 – Proof of Concept

Phase 3 – Std Design Proc. And Auto-FEA

Phase 4 – Database Manager



Define

CATERPILLAR®

Current State

Current Process
Running Below 6
Sigma Capability

214 User Inputs

346 Reported Values

45 Designer Checks

Current Design Tool created 10 years ago and newer tools were not being added

New requests coming in were done over the phone or in meetings and not tracked

Multiple people involved in the design process. (Designers and Analysts)

6 Manufacturing facilities with 3 different design centers



Define/Measure

CATERPILLAR®

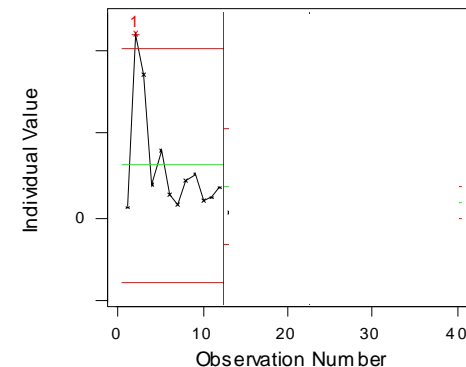
**Goal:
Meet/exceed
customer
expectations**

Problem: Uncontrolled Design Process

- No process control between designers
- Historical information kept by each designer
- Overall design time not meeting customer requirements

Goals:

- Implement standard design process
- Reduce time required for design
- Reduce product costs
- Maintain/Increase product quality





Define



Partnered with MSC.Software to define Project

Phase 1 – RDA

Phase 2 – Proof of Concept

Phase 3 – Std Design Proc. And Auto-FEA

Phase 4 – Database Manager (MSC.SimManager)

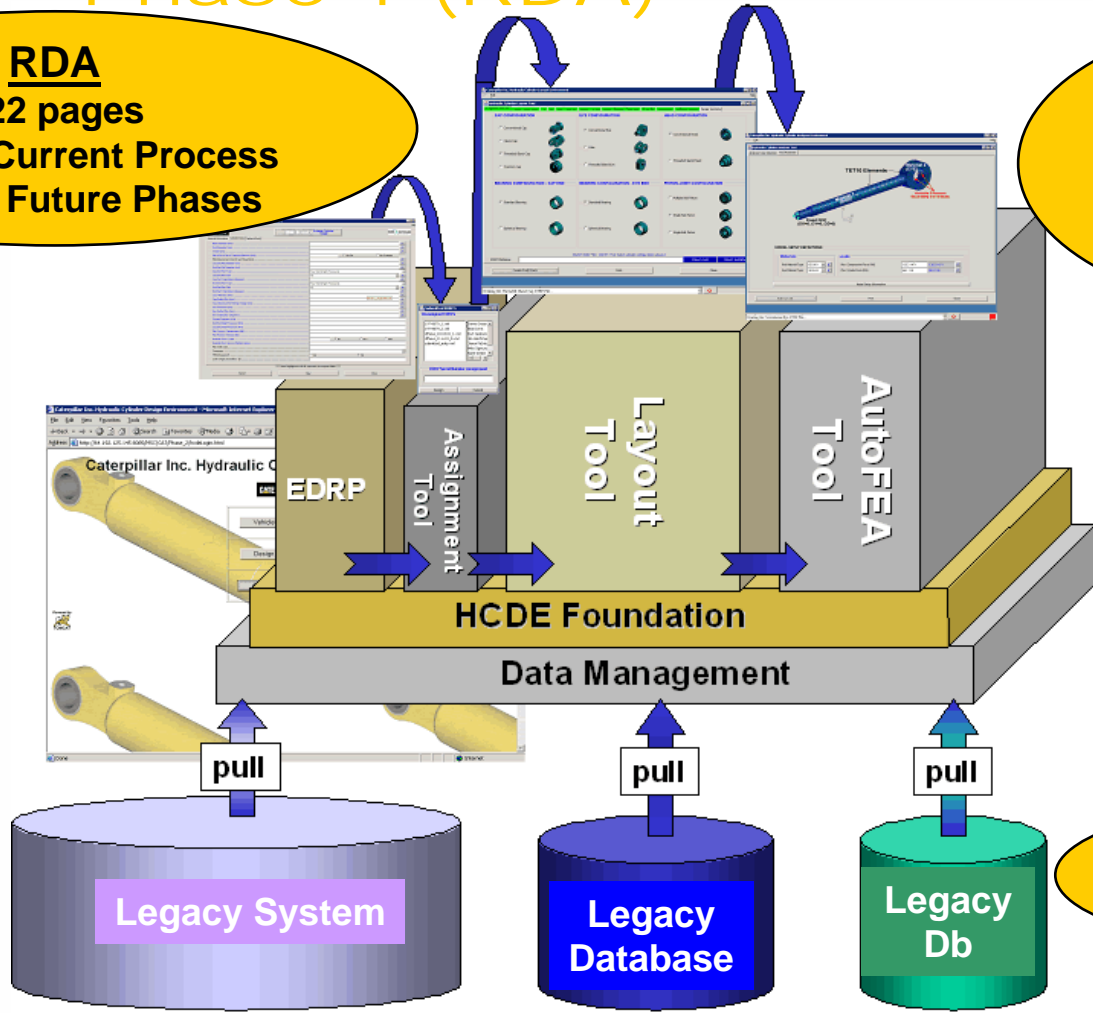


CATERPILLAR®

Define - Phase 1 (RDA)

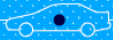
RDA
222 pages
Outlines Current Process
Outlines Future Phases

The RDA laid out the framework for the design environment and the future phases.



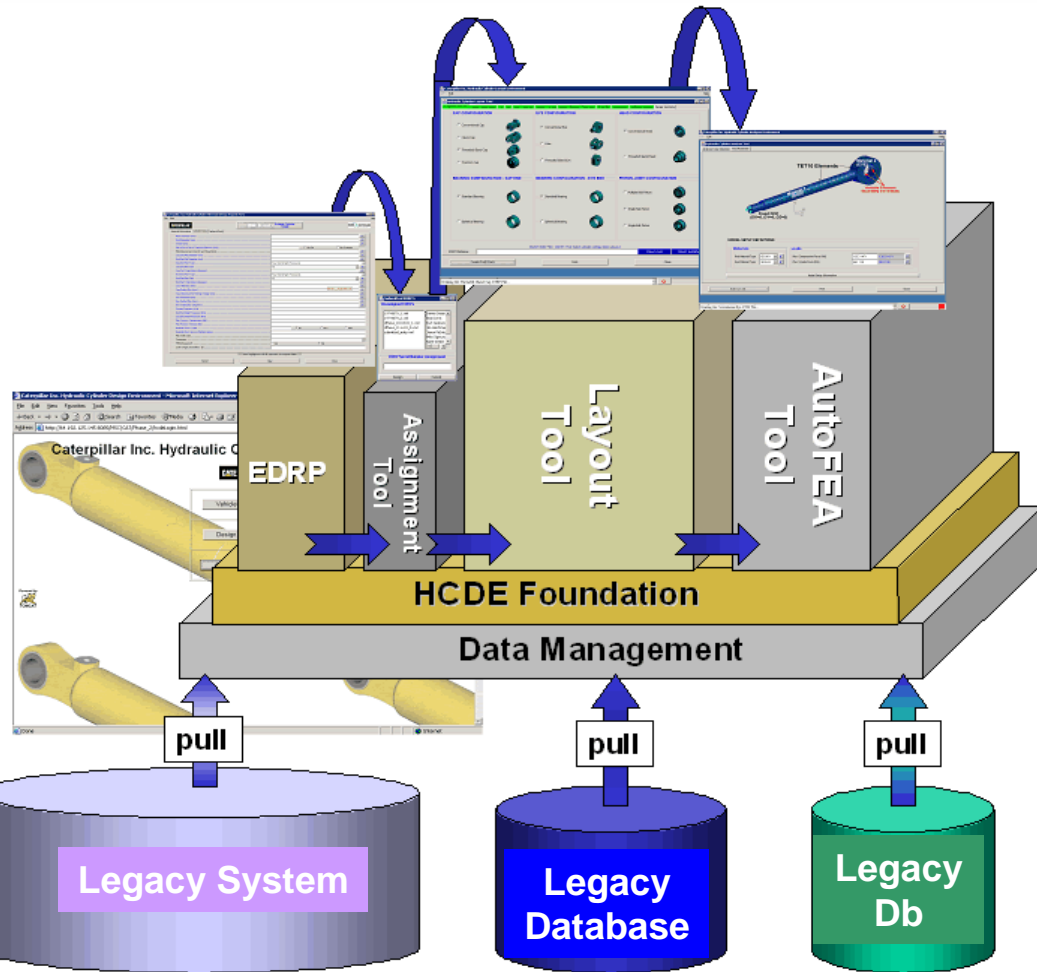
Critical Customer Requirements identified

PRODUCT DEVELOPMENT CONFERENCE



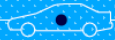
CATERPILLAR®

Define/Measure – Phase 2 (Proof Of Concept)



- Measured “As-Is” Design Process
- Completed overall structure of Electronic Design Request Form (EDRF)
- Proof of concept on automating FEA
 - Java GUI for Inputs
 - Automated processing of FEA with results
 - MSC.Patran

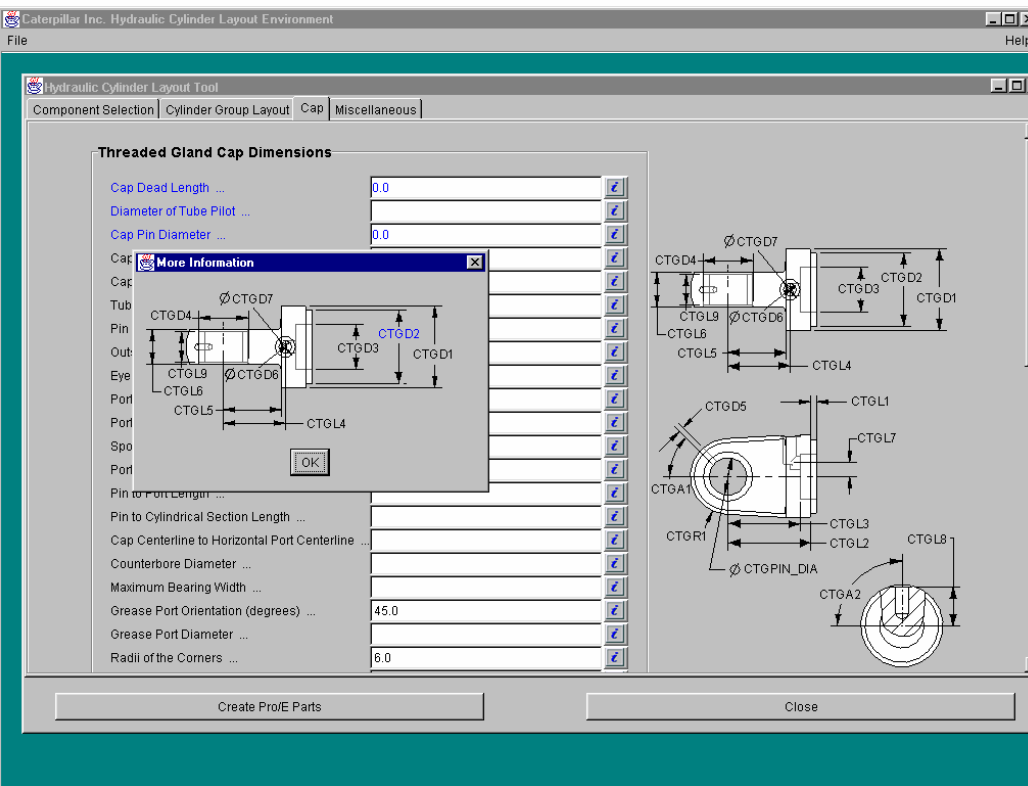
Measured results showed significant savings potential



Measure/A/I – Phase 3 (Production Release)

- Complete web based GUI with a structured design process
 - Guides users from initial request through the completed design
- Automatic FEA with no analyst required for standard geometry
- Integrated legacy systems into one design environment
 - Designers have one safe source for all necessary data
- Integrated conversion process for 3D modeling
 - MSC.Patran and STEP conversion

Measured results showed an 80% reduction in design time



Process is now statistically under control

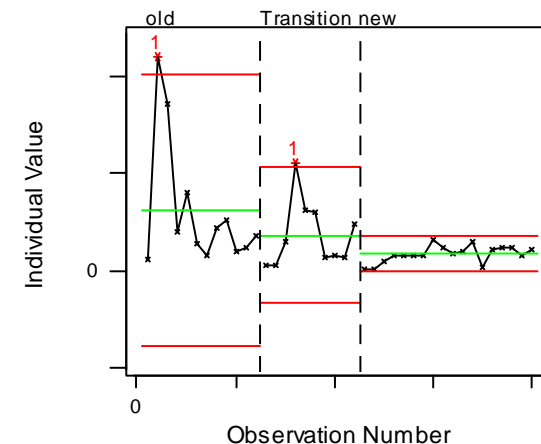


Measure/A/I/C – Phase 4 (Production Release)



- Updated web based portal for server side processing
- Integrated more legacy systems for a more robust design process
- Implemented MSC.SimManager database manager
 - Design History retained between designers
 - Simplifies reuse of current components

I Chart for Time to by Process



\$\$\$\$

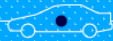


CATERPILLAR[®]

Return on Investment

Repaid Initial investment through Phase 3 in First Year

- Avoidance of hiring new Analysts
- Avoidance of hiring new cylinder designers
 - Increased capacity of current design and analysis staff
- Does not include the avoidance of warranty that will be seen in the next few years



Application of Simulation in Hospital Bed Development

William Olson

Senior Mechanical Simulation Engineer

Roberta Murnyack

Advanced Biomedical Engineer

Hill-Rom Co., Inc.

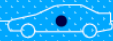
PRODUCT DEVELOPMENT CONFERENCE



Abstract

Hill-Rom
A HILLENBRAND INDUSTRY

Until recent development, simulation techniques have had only targeted application within the hospital bed and accessory market. With requirements for these products becoming more sensitive to weight and care giver safety, this application must be significantly expanded. Additionally, driving the use of simulation is a more competitive market place. New products must realize commercialization sooner than ever before. Simulation not only provides benefits from an engineering and performance perspective, but reduces development costs and time to market. As an extension to simulation, automation can greatly increase efficiency. This presentation describes simulation usage in this environment and the return on investment which may be possible.



Introduction

Hill-Rom
A HILLENBRAND INDUSTRY

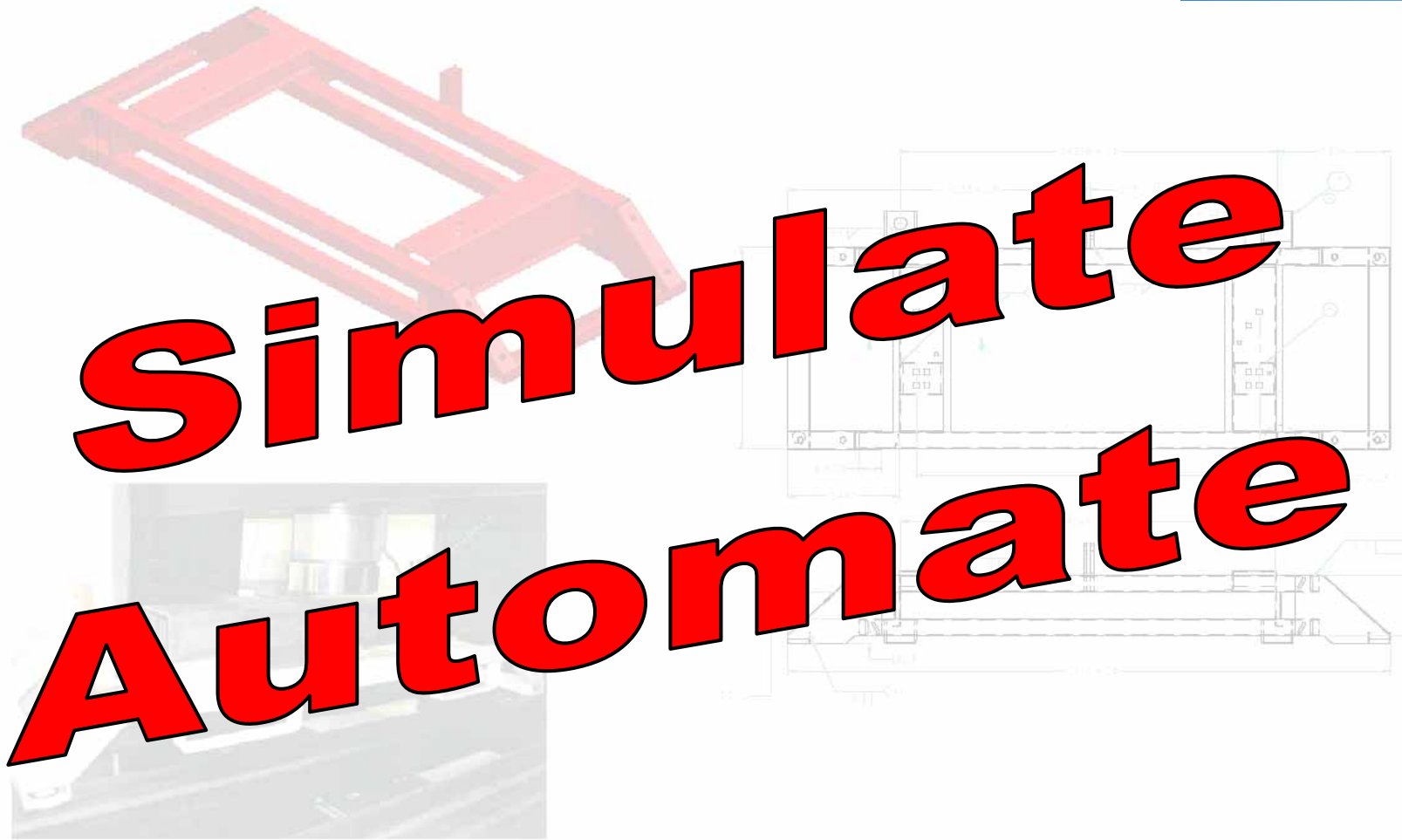
Hill-Rom helps healthcare professionals around the world create safer, more efficient and clinically-effective patient care environments and processes that improve the overall quality and cost of care delivered to patients in acute, long term and home care environments. One facet of this is through the design and manufacture of medical devices which include hospital beds and other patient care systems. A focus of research and development has been on improving care giver safety. This is realized in more simple care giver interfaces, decreased actuation forces, and significantly lighter products.

Like other industries, Hill-Rom is sensitive to market pressures. It is critical that we provide the market with useful products in accelerated development cycle times.



Problem Definition

Hill-Rom
A HILLENBRAND INDUSTRY

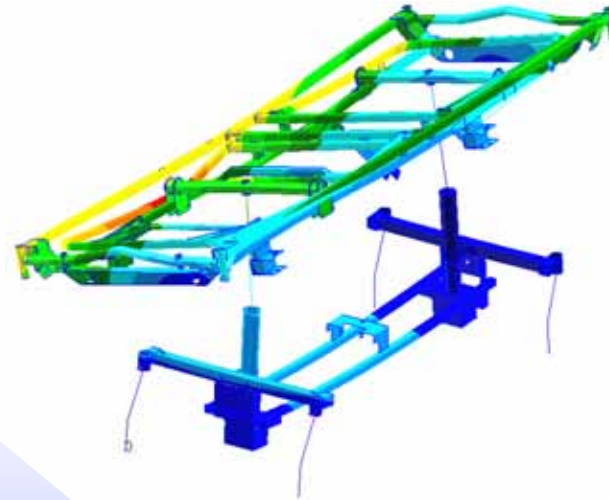


simulate
Automate

PRODUCT DEVELOPMENT CONFERENCE

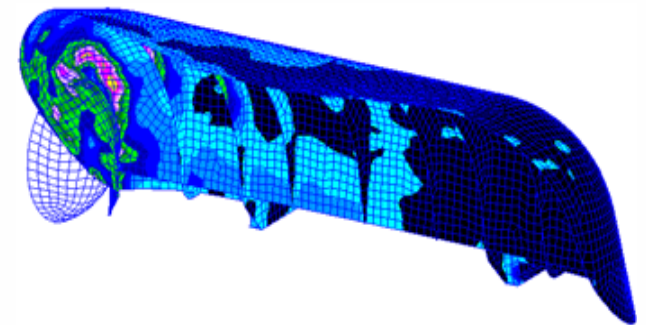
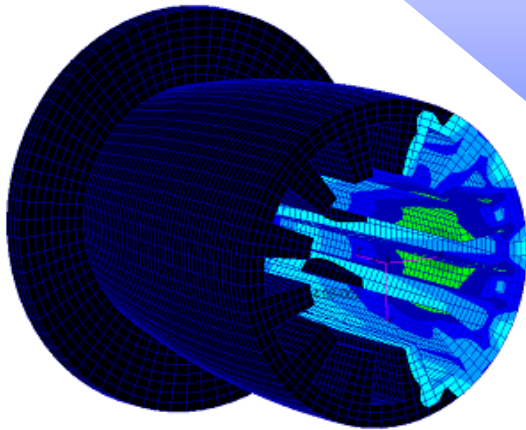


Simulation



Linear Modeling

Nonlinear Modeling



Dynamic Modeling

PRODUCT DEVELOPMENT CONFERENCE




Automation

Automation can extend the benefits of simulation through modeling efficiency gains. Automation can be accomplished through a custom developed environment



Hill-Rom Simulation Workflow

Welcome to the Hill-Rom Structural Analysis Medical Quality Engineering Surfaces package.



These pages are maintained by the simulation group.

Please direct all technical questions or problem reports to the Bill Olson at bill_olson@hill-rom.com

- [Start Modeling](#)
- [Help](#)

Create Foam/Ribbed

Geometry is created using user inputs. Please enter your inputs.

Input Cell Width: Please enter the cell width (inches) in the box below.

Inputs

Cell Width (y-coordinate): 5

Cell Length (x-coordinate): 36

Rib Height (z-coordinate): 2

Foam Width (y-coordinate): 27

Foam Length (x-coordinate): 44

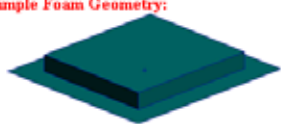
Foam Height (z-coordinate): 5

MSC.Acumen

Apply Inputs

Back Step

Create Simple Foam Geometry:



Geometry is created using a list of parameters that you, the user, inputs. Please enter your parameters and select **Apply Inputs**.

Input Length: Please enter a value for the bladder Length (inches) in the box below.

Input Width: Please enter a value for the foam height (inches) in the box below.

Inputs

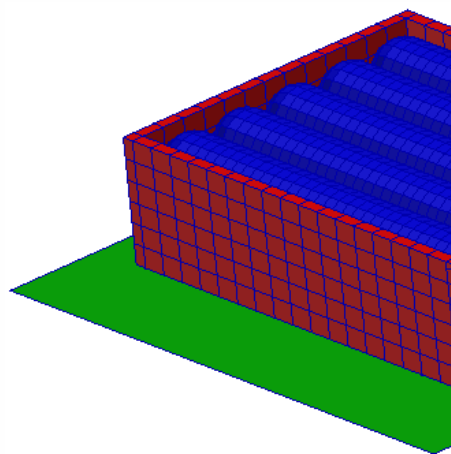
Length (y-coordinate): 40

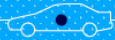
Width (x-coordinate): 36

Height (z-coordinate): 5

Apply Inputs

Disc Step





Return on Investment

Hill-Rom
A HILLENBRAND INDUSTRY

This ROI example is based on a recent effort to develop a simulation process for a major subsystem and automate it within an easy-to-use tool. After a basic process cost comparison is completed, four ROI models are evaluated

- Net Present Value (NPV)
- Internal Rate of Return (IRR)
- Break Even
- Intangibles



ROI-Baseline Process Cost Evaluation



A Design Structure Matrix (DSM) approach is utilized to determine both the baseline cost and the improved process cost.

Baseline Process

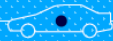
	1	2	3	4	5	6	7	8	9	10	11
Identify Requirements	1	0	0	0	0	0	0	0	0	0	0
Design Concept Model	2	1	0	0.8	0	0	0	0	0	0	0
Build Physical Concept Prototype	3	0	0.8	0	0	0	0	0	0	0	0
Perform Physical Testing - Concept Model	4	0	0	1	0	0	0	0	0	0	0
Incorporate Design Changes	5	0	0	0	0.75	0	0.75	0.75	0.75	0	0.75
Build Physical Engineering Prototype	6	0	0	0	0	0.8	0	0	0	0	0
Perform Physical Testing - Engineering Model	7	0	0	0	0	0	1	0	0	0	0
Perform Full Body Physical Testing	8	0	0	0	0	0	0	1	0	0	0
Verification of Engineering Model	9	1	0	0	0	0	0	0	1	0	0
Build Physical DVM Prototype	10	0	0	0	0	0	0	0	0	0.8	1
Perform Physical Test - DVM Prototype	11	0	0	0	0	0	0	0	0	0	1

2) Assign rework dependencies

1) Define Process Tasks

3) Assign total number of cycles through process

4) Assign \$ values to get total costs

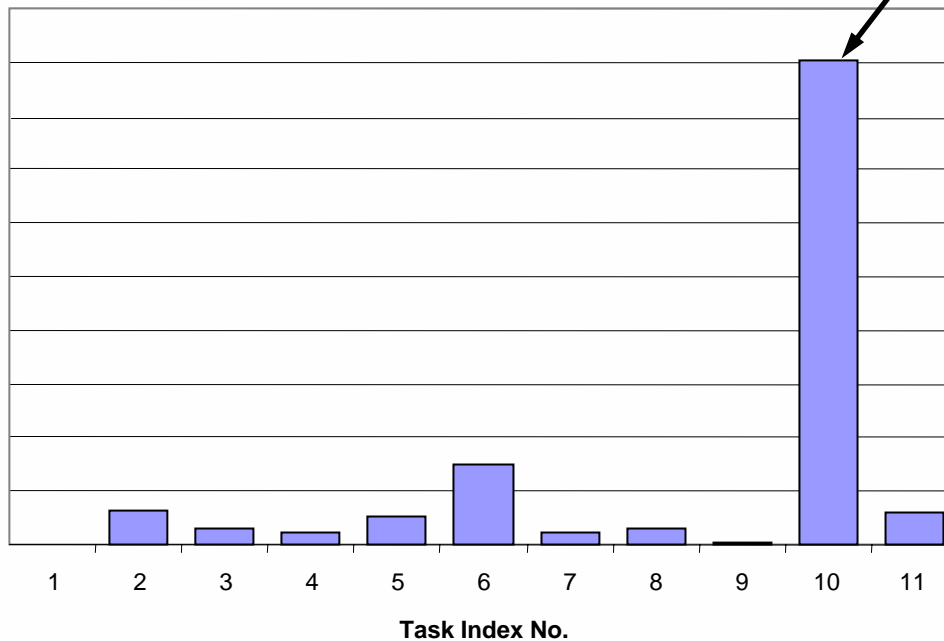


ROI-Baseline Process Cost Evaluation



*Per Task Cost Evaluation

Total Process Cost



“Build Physical DVM Prototype” represents the majority of the cost of development

*Includes all iterations



ROI-Improved Process Cost Evaluation



Improved Process

		1	2	3	4	5	6	7	8	9	10
Identify Requirements	1	1	0	0	0	0	0	0	0	0	0
Design Virtual Concept Model	2	1	1	0	0	0.75	0	0	0	0	0
Analyze Virtual Concept Model	3	0	0.8	0	0	0	0	0	0	0	0
Incorporate Design Changes	4	0	0	0.75	1	0	0.75	0.75	0.75	0	0.75
Build Physical Engineering Prototype	5	0	0	0	0.1	1	0	0	0	0	0
Perform Physical Testing - Engineering Model	6	0	0	0	0	0.1	1	0	0	0	0
Perform Full Body Physical Testing	7	0	0	0	0	0	0.1	1	0	0	0
Verification of Engineering Model	8	1	0	0	0	0	0	0.1	1	0	0
Build Physical DVM Prototype	9	0	0	0	0	0	0	0	0.1	1	0
Perform Physical Test - DVM Prototype	10	0	0	0	0	0	0	0	0	0.1	1

Perform same type of assessment for the improved process

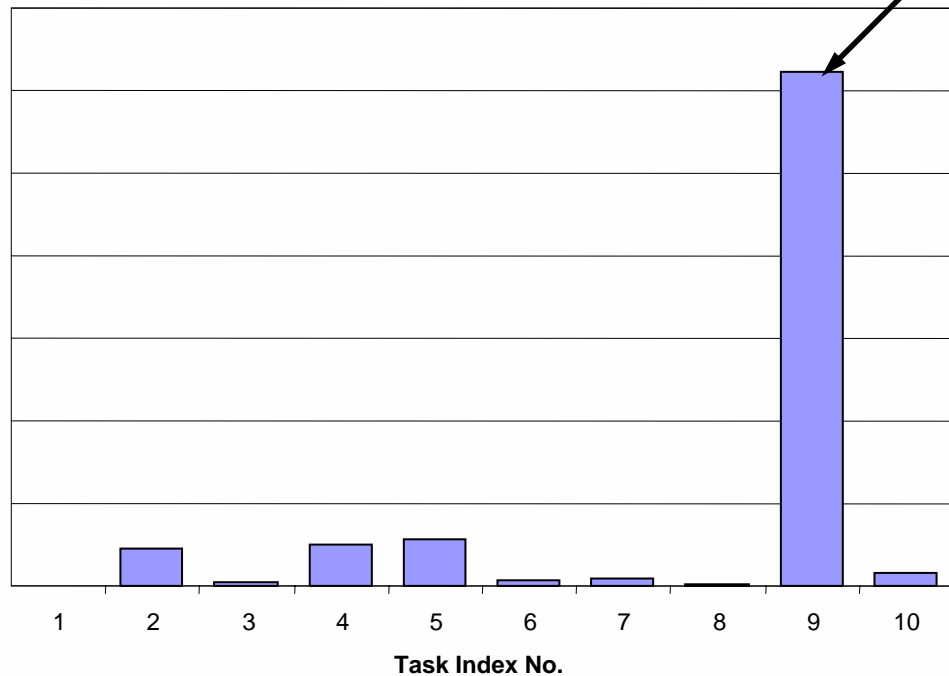


ROI-Improved Process Cost Evaluation



*Per Task Cost Evaluation

Total Process Cost



“Build Physical DVM Prototype” represents the majority of the cost of development. However, it is significantly (32%) smaller than the baseline process

*Includes all iterations



ROI-Process Cost Comparison

Hill-Rom
A HILLENBRAND INDUSTRY

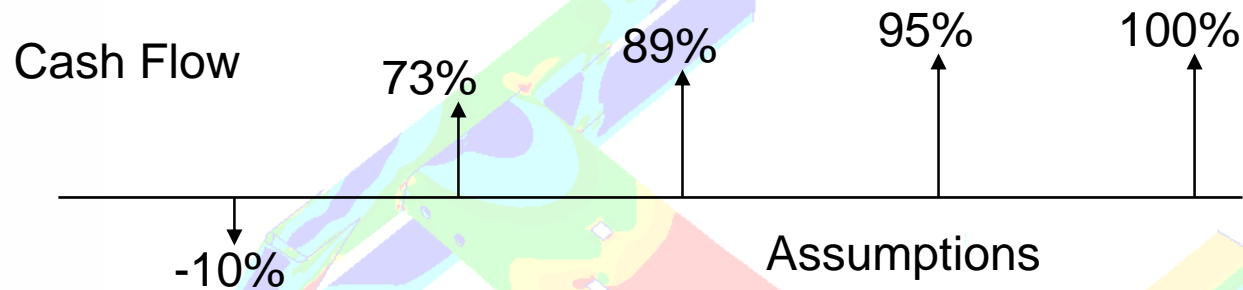
For a fully mature process comparison, the cost evaluation predicts:

- a reduction in cost of **48%**
- a reduction in time of **38%**



ROI-Tangible Evaluations

Hill-Rom
A HILLENBRAND INDUSTRY

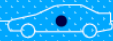


Assumptions

- Graduated Implementation
- 2 design projects/year
- 20% maintenance/year
- Percentage of Entitlement

- NPV-\$403K Improved Process (15% hurdle)
- IRR-731% return over 5 year implementation timeframe
- Break Even-14 months

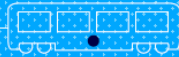
PRODUCT DEVELOPMENT CONFERENCE



ROI-Intangible Benefits

Hill-Rom
A HILLENBRAND INDUSTRY

- Increased development opportunity.
- Intimate knowledge of full field behavior, not just point wise information.
- Repeatable evaluation (boundary condition normalization).
Some systems are subject to variation of boundary conditions which, from a test environment, are not known. This variation, while necessary to understand, complicates a nominal design assessment.
- Some new performance measures, while not verifiable are realized, and may be used for alternative comparisons.

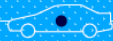


Conclusions

Hill-Rom
A HILLENBRAND INDUSTRY

- Simulation techniques can make vast improvements in the process for developing hospital beds and accessories.
- Automation of this process can further increase productivity and allow for increased cycles of learning.
- Application of this tool has the potential to greatly improve product performance through a more thorough understanding of the product.





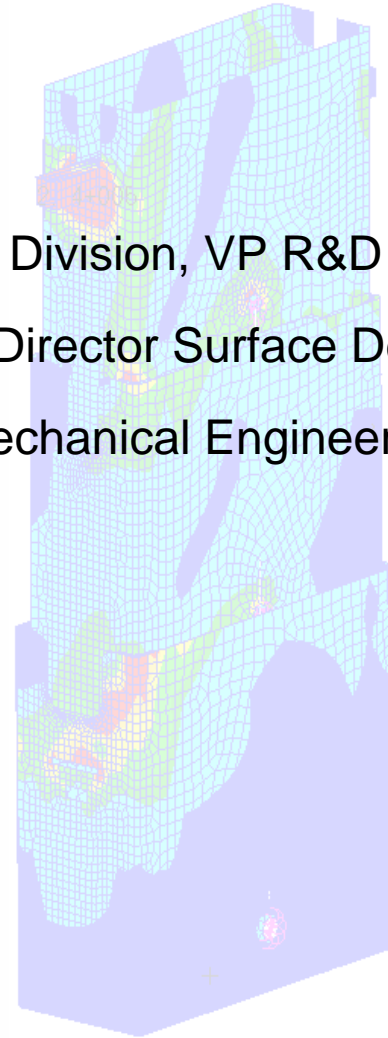
Acknowledgements

Hill-Rom
A HILLENBRAND INDUSTRY

Dave Ball, Hill-Rom Co., Inc., Clinical Division, VP R&D

Jim Stolpmann, Hill-Rom Co., Inc., Director Surface Development (Retired)

Eric Meyer, Hill-Rom Co., Inc., Sr. Mechanical Engineer





References

Hill-Rom
A HILLENBRAND INDUSTRY

Design Structure Matrix (DSM) method, Steward 1981;
Smith and Eppinger 1997; Cronemyr, Ronnback and
Eppinger 2001 (ref. <http://www.dsmweb.org>).



Centralized Electronic Licensing

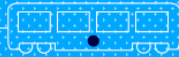
Ron Ellis

Northrop Grumman Corporation

Project Manager, Multi-sector License Sharing Project

MSC Software Virtual Product Development Conference

Huntington Beach, California October 18-20, 2004



Outline of Presentation

Abstract

Introduction and background

License sharing in a highly distributed environment

What is CELUG?

Conclusion



Abstract

This presentation is a case study of the initiative between Northrop Grumman and MSC.Software Corporation to establish a methodology that enables a diverse number of users to share engineering software tools in a multi-sector environment. In particular, a methodology for allocating bill back charges in a distributed enterprise is presented. Also included in the presentation is a review of the best practices among members of the centralized electronic license user group (CELUG) organized in January of 2003. Membership in the CELUG organization is represented by over 25 major US companies and universities. The CELUG goal is to share best practices, help consult other companies that are considering centralized licensing, discuss issues that are affecting deployment, and present to vendors with one voice regarding requirements, functionality of Macrovision FLEXIm software and daemon updates, options, etc.



Introduction and Background

Software Asset Management Challenges

- Measuring license utilization
- Existing license management tools do not provide the necessary features to measure business value
- FLEXIm is the standard tool used by many Electronic (EDA) vendors

License Sharing Initiative at NGC

- Existing As-Is environment
- Pilot exercise to validate license sharing feasibility
- Prerequisites to license sharing practice

Requirements for Improving License Management

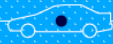
- Dialog is needed among ISV's, Macrovision and End Users
- Standards are needed for licensing systems, software licensing vendors' licensing implementation, analysis solutions requirements, product-to-feature mapping, and certification organizations
- Access to complete, accurate data from the licensing system
- ISVs need standards throughout their own development environment all the way through their sales and customer fulfillment system, so that the products created are consistent with the products sold and therefore with the features cut for the license files



Multi-Sector License Sharing Initiative

Objective – Enable enterprise-wide sharing of MSC Licenses

- Consummate an Enterprise Agreement with MSC to convert MSC Licenses (Floating and Node Lock) to MSC.MasterKey Licenses
- Reduce MSC maintenance cost
- Provide more flexibility to users
 - ✓ Potentially avoid future buys
 - ✓ Provide easy access to New tools
 - ✓ Higher utilization of MSC.Software resources
- Productivity gains for engineering
 - ✓ Achieves productivity improvements by the use of common software tools and an expanded resource pool
 - ✓ Streamlines processes for tracking / administration of MSC and other engineering software licenses
 - ✓ Improve accuracy of determining software requirements (via our improved visibility into actual software usage)
- First step in overall Engineering license sharing strategy (Autocad, CATIA, Unigraphics, Metaphase, Pro-Engineer, TeamCenter, etc.)
- First step in overall hardware sharing strategy



Types of Licensing Models

Flat-rate model

- One flat rate per person, per company
- Based on headcount regardless of type of headcount
- Example: sun java enterprise system

Utility-based model

- Modeled after the utility industry (electricity, phone, etc.)
- Pay only what is used, when it's used (pay per use)
- Examples: IBM, HP, EDS

Subscription-based model

- Modeled after the publication industry
- Subscribe for a duration (week, month, 3 months, etc.)
- Example: Symantec, McAfee

Token-based model

- Check in/out licenses based on token count
- Products/features available via a token pool
- Example: AVEVA, Accelrys, Agilent, MSC.MasterKey system



MSC.MasterKey License Overview

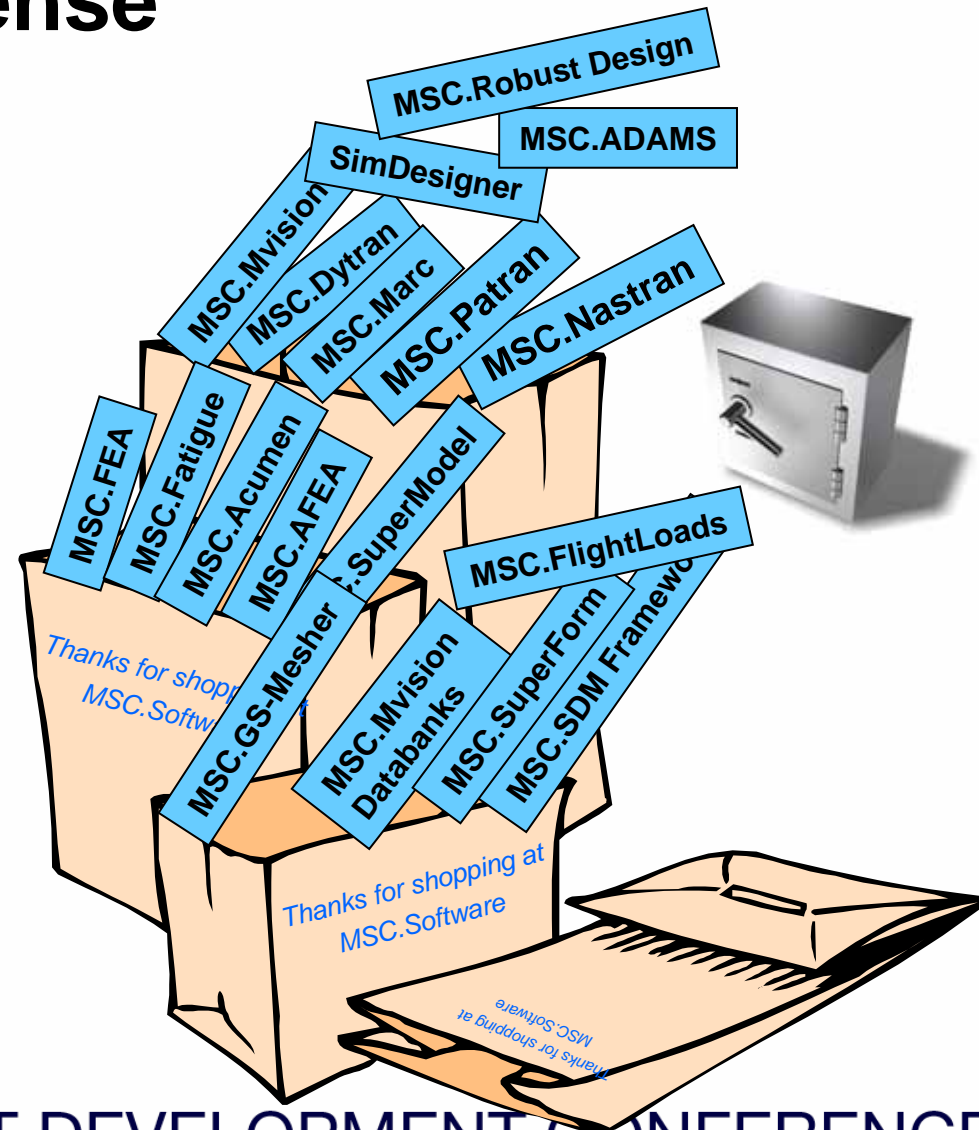
A single license provides access to breadth & depth of MSC.Software's portfolio of over 250 items

Based on industry standard licensing system (FLEXlm)

Utilizes MSC's exclusive token-based system

With sufficient tokens, no usage restrictions in frequency, number of seats, or simultaneous checkouts

Facilitates license sharing in a distributed environment





MSC.MasterKey License Overview Cont.

How the token-based method works

- Each individual product feature requires a certain number of tokens to run
- Tokens are checked out from the pool on the license server
- Tokens enable the software feature(s) needed
- Job is performed
- Tokens returned to pool for other use

When all tokens are checked out

- Batch request goes into queue
- Interactive request – first come/first served basis



What Is CELUG?

Centralized electronic licensing user group (CELUG)

- Organized in January 2003 to promote collaboration among ISV's, MacroVision and end-users regarding the use of FLEXIm and the need for better software asset management tools
- CELUG's mission is to share information and experiences on deploying centralized licensing systems.
- Currently there are over 65 members representing ~25 companies
- The web site can be accessed at <http://www.celug.Org>

CELUG objectives for 2004

- Publish a "white paper" on CELUG's position on software licensing
- Participate in the 2004 SoftSummit

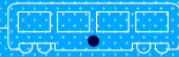


Conclusion

A token-based flexible software licensing approach, such as MSC's MasterKey system, helps to address a multitude of virtual product development challenges

- **Accessibility**
- **Breadth and depth**
- **Flexibility**
- **Simplicity**
- **Cost effectiveness**
- **Manageability**
- **Supplier consolidation**

The future of VPD software resource management may include innovations such as detailed usage tracking and reporting, and alternative pricing models such as pay-per-use



Just In Case



Methodologies used to measure usage - reports used to analyze costs, usage, performance, access response, and sites getting licenses?

- 1. Daily report created from FLEXlm licenses: information is gathered from the lmgrd log files. The report is used to determine the number of denials and if additional software licenses are required. Denials are reported for last 24 hours for available licenses checked out and for modules not available in the license file.**
- 2. Hourly % usage in FLEXlm licenses: an automated reporting script looks for % usage in FlexLm licenses and sends usage status for licenses which are being used at the 90% or higher capacity. This report is run 15 minutes past each hour using lmstat. SAM alarm provides similar reports.**
- 3. Monthly usage over time graph: this graph shows the overall usage trend for each of the module over time.**
- 4. Monthly usage summary report: this is the usage summary report listing the total number of hours software is being used/available. It also provides a break down by individual user and information such amount of time used, user platform (hardware/OS), IP address. It also includes user identification information from the employee database, such as user name, dept number, business unit/division and country of location.**
- 5. Users using the interactive software for over 24 hours: this report is actually a reminder to users who have checked out a license for interactive applications for over 24 hours. This is done by lmstat, looking at the license age and then looking the user ID in the employee database.**



Methodologies used to measure usage – cont.

6. Daily denials reports for the previous 24 hours with automated emails to the appropriate team members for each vendor. This process analyzes “roll-overs” (where the user rolls over to another server and is able to obtain a license there) so that they can be distinguished from “true denials”.
7. Current usage, for certain "watch list" tools every 8 hours, and all tools every 24 hours. If usage for a tool is over 80%, provide automated emails that go out to any users of that tool who have had a license checked out for an extended period of time. After 3 notifications, the user's manager and EDA site contact are automatically updated.
8. Monitor peaking licenses on a daily basis. Weekly summary report which shows all of the peaking licenses with usage of over 25% for the current month.
9. Database of license users. Use this database to get the info for sending out automated emails, and to post-process user specific data.
10. SAMreports of monthly high water mark, usage efficiency, usage over time, usage summary by user, usage summary by feature, and usage summary by location. The location report is produced using the \$LM_PROJECT variable set to each user's DNS domain via an end user configuration script. All of these reports are updated nightly to show up-to-date info, and are available on the web for all users.
11. SAMreports specifically requested by vendors include a quarterly usage summary by feature report, to analyze usage over a longer period of time.