Critical Mass: Estimating software size from use case points and requirements repositories

Lee Fischman

Galorath, Incorporated

and

Dan Ferens

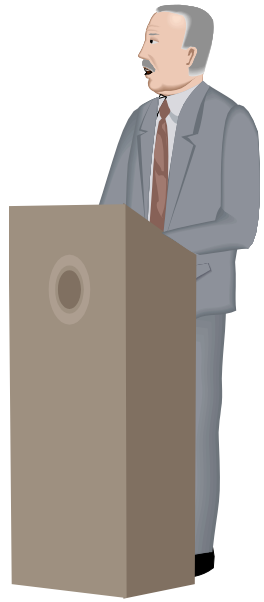
Air Force Research Laboratory

18th COCOMO/SCM Forum

University of Southern California

October 21, 2003

Presentation Outline



- Project Overview
- Critical Mass Estimating Objectives
  - Use Cases
  - Requirements Repositories
- Program Progress
- The Future

GALORATH

Project Overview

- Project awarded to Galorath Incorporated in June 2002 to develop a line of tools for automatically estimating software size from development documents
  - Awarded by the U.S. Air Force as part of the Small Business Innovation Research (SBIR) program
  - Phase 2 award following a successful Phase 1
  - Two-year project, approximately $750,000
  - End product: standalone tools interfacing with industry-leading specification and requirements development tools (Rational Rose, Telelogic DOORS)
- Requires development of two closely related tools, along with data collection and estimating method enhancement
  - Unified Modeling Language (UML) diagrams
  - Structured lists of requirements

GALORATH

History of Software Sizing

- Pre 1986 Primitive Methods (E.g. words of memory)

- Late 1980's: SLOC and Function Points
  - Estimating methods available
  - Limitations to size artifacts (SLOC & Function Points)

- Late 1990's: Object Counts
  - Several methods available, primarily based on use case points
  - Limitations to object counts: different definitions, limited application, design legacy (new, reused, etc.)

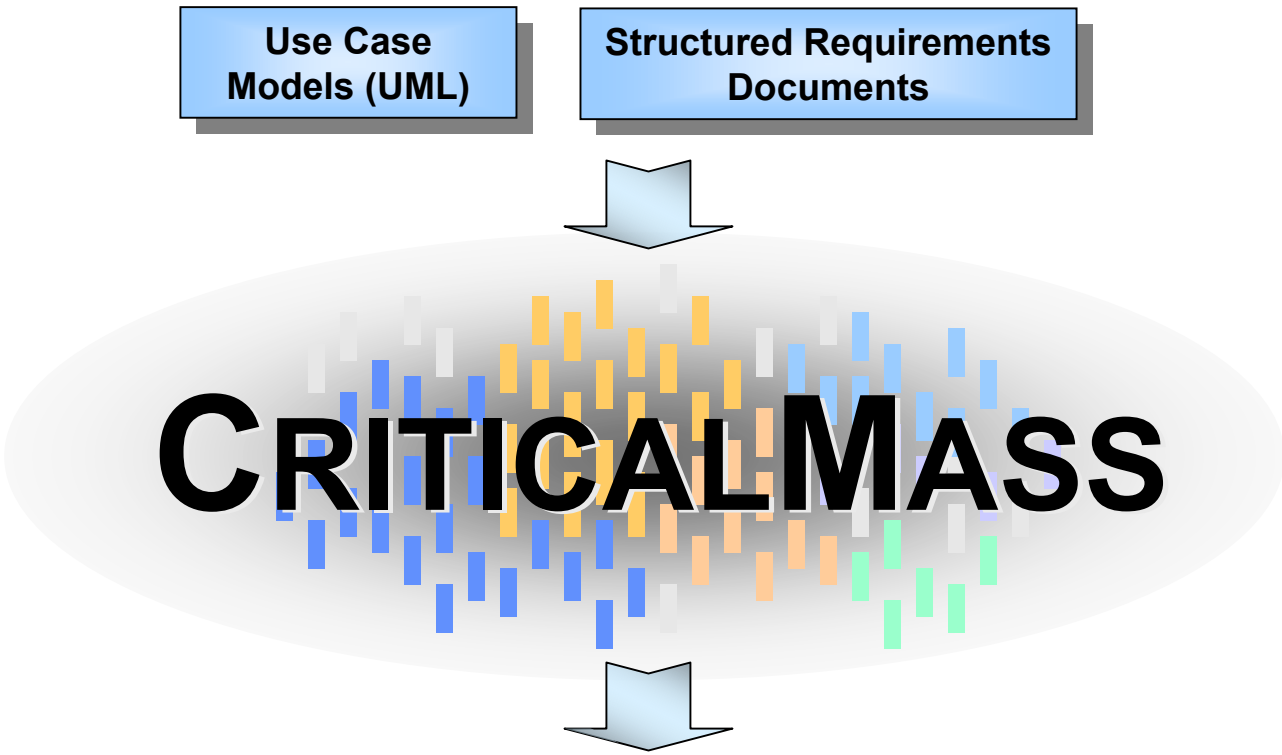
Bottom line.. Size still needs research

Therefore this SBIR

GALORATH

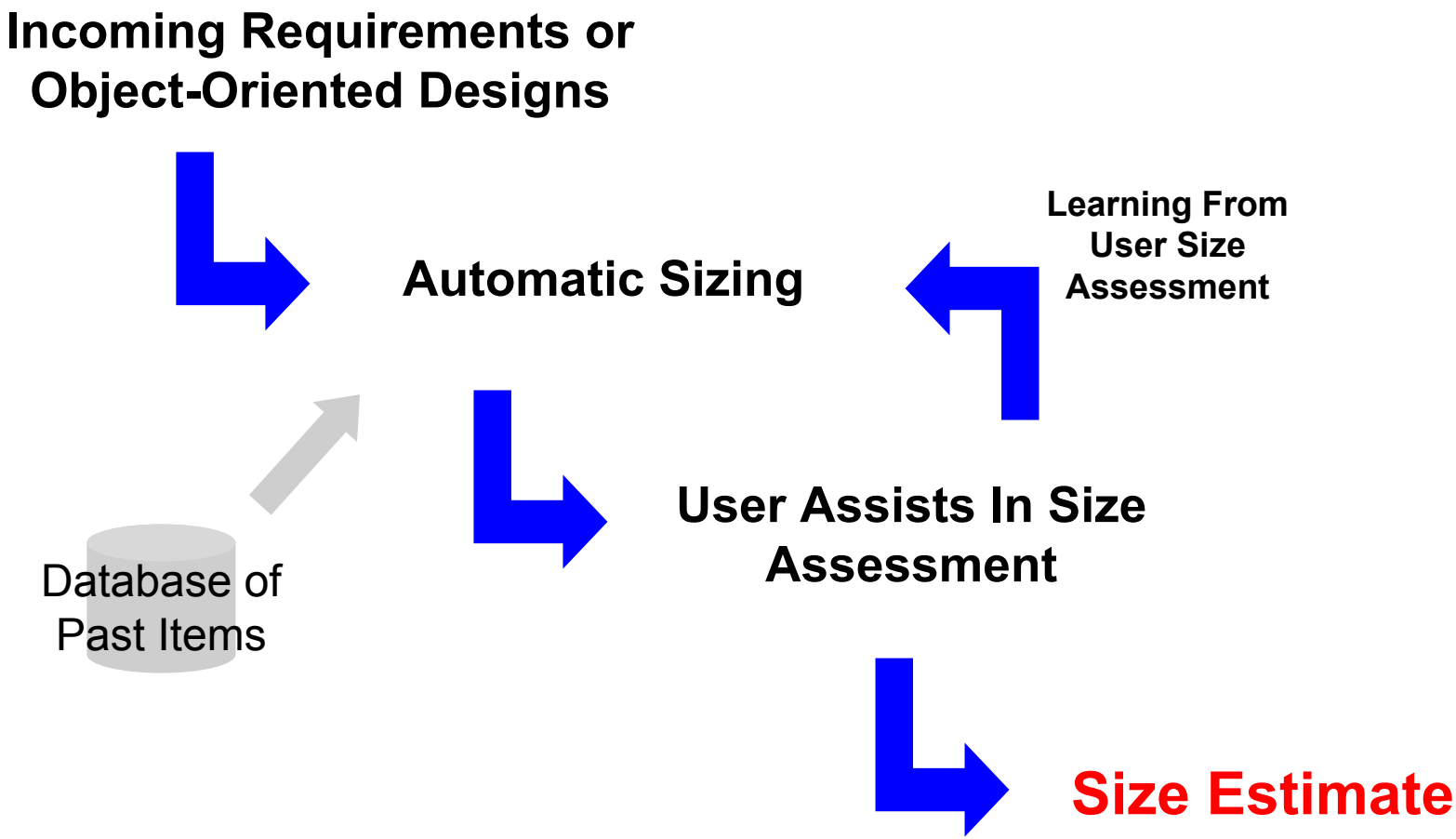
An Automated Sizing Toolset

From Software Design & Requirements Artifacts…



GALORATH

CriticalMass Work Flow



Estimating Size Based On Use Cases

- Use Case: notional description of a system
  - Frequently used at the earliest stages in a project
- Use cases have three descriptive characteristics which can be exploited to provide sizing information:
  - Number of use cases
    - Use cases provide a high level description of the intended function of the system. A small application might have only one use case, while very large applications may have hundreds.
  - Number of “actors” per use case
    - Actors are the “agents” that interact with the software and so, use cases must have at least one actor. A commonly accepted standard is one actor per use case (more than one actor means that there must be more than one use case).
  - Number of scenarios
    - Potential outcomes of the software
    - No limit to number of scenarios in a use case

GALORATH

The Normalized Use Case (NUC)

- Challenge - A “standard” use case may be quantifiable using a given software metric, but none are standard.

- Theory – The components of a use case are interrelated, so the “sum of the whole” may equal more than the “sum of its parts”.

Whereas:

Use Case Points = sum(Actors) + sum(Use Cases)

for all use cases <u>combined</u>

Normalized Use Cases = Actors + Relations + (Actors * Relations) + …

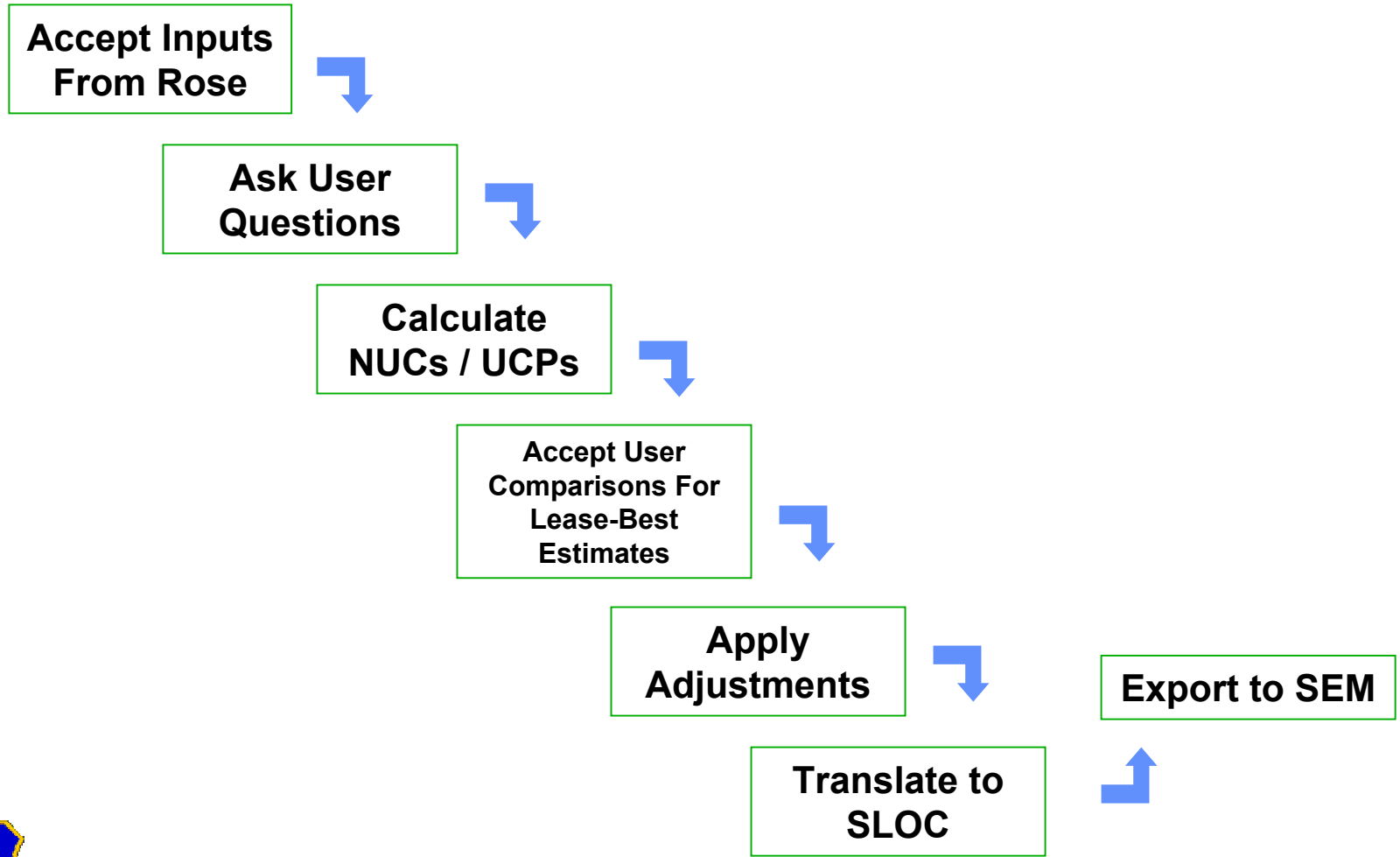
for <u>each</u> use case

- A Normalized Use Case (NUC) counts and cross-multiplies all artifacts:
  - **Actors**
  - **Unassociated Relations**
  - **Includes**
  - **Extends**

- By using many terms, the NUC may be robust to varying information.

GALORATH

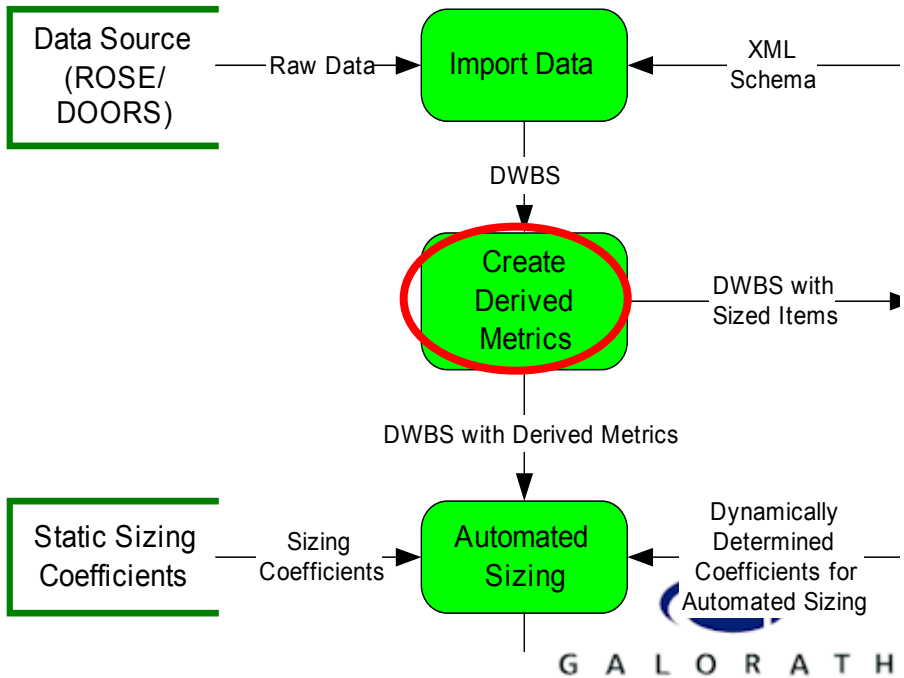
UML Tool Flowchart



GALORATH

Sizing From Requirements Documents
Create Derived Metrics - DOORS

- **Number of requirements linked to a WBS item**
- **Word count**
- **Information density (using compression algorithms)**
- **Source documents in modules**
  - **(Word, Excel, other)**
- **Document length**



Dynamic Coefficient Determination

Methods For Calibrating the Sizing Algorithms

- **Inputs are sized items with derived metrics: from a repository, from the existing project, or items sized by the user (via size by comparison)**

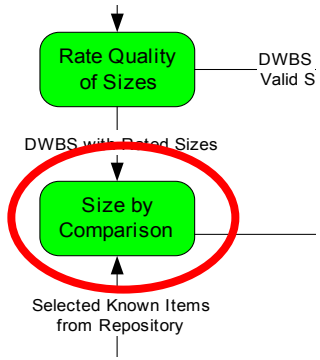
- **Methods being used:**
  - **Least Average Deviation**
  - **Least Squares Regression**
  - **Simulated Annealing**

- **Default coefficients will be used when insufficient data is available for calibration**

GALORATH

Sizing Aids: Size By Comparison

Augments Automated Sizing

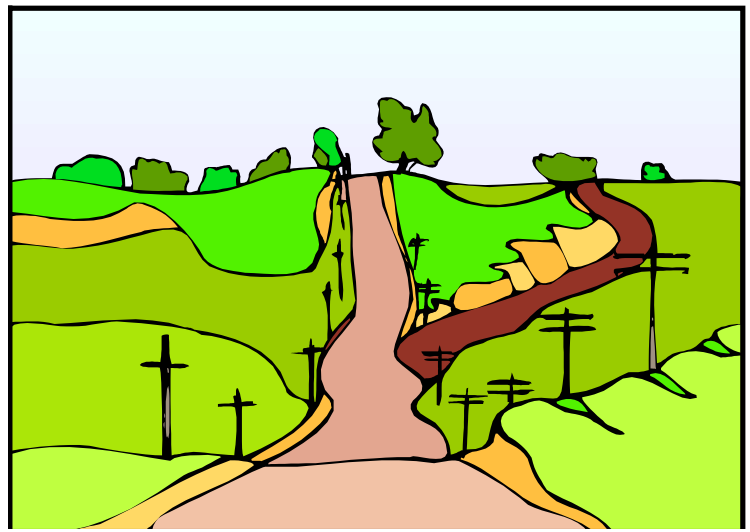


- Lets the user provide input about least understood items
- Entries are made via the user-friendly method of pair-wise comparisons
- Uses the proven Analytic Hierarchy Process (AHP) algorithm

GALORATH

The Road Ahead

CRITICAL MASS



- Rose and DOORS Tools Already Integrated With Respective Products
- Estimating Methods Being Integrated
- Beta Versions Ready by Q1 ‘04
- Project Completed by 1 April

GALORATH