Current and Future Challenges for Ground System Cost Estimation

Barry Boehm, Jim Alstad, USC-CSSE

GSAW 2014 Working Group

Session 11F –

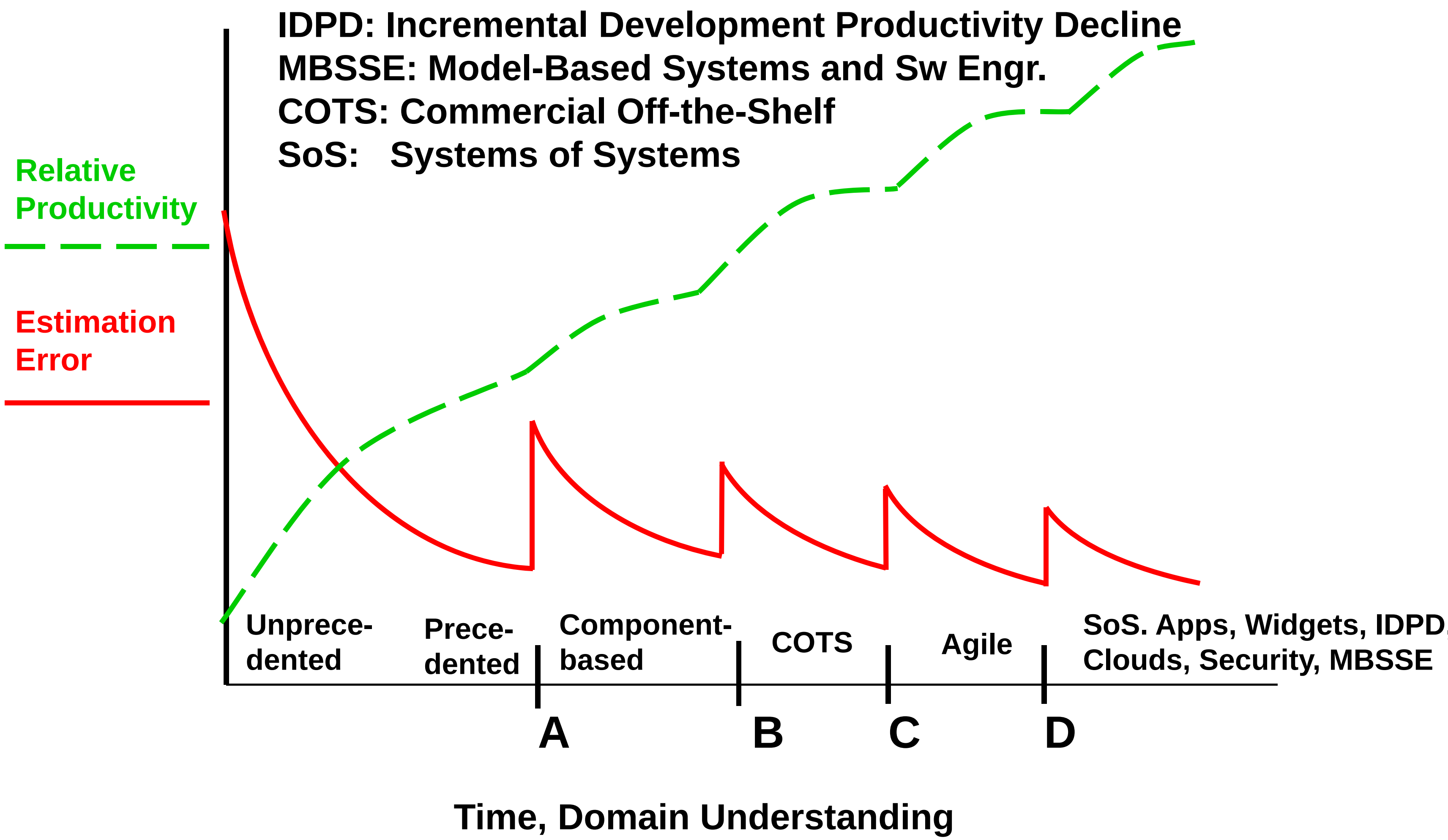
Cost Estimation for Next-Generation Ground Systems

February 26, 2014

Summary

- Current and future trends create challenges for ground system cost estimation
  - Mission challenges: emergent requirements, rapid change, net-centric systems of systems, COTS, clouds, apps, widgets, high assurance with agility, multi-mission systems
- DoD Systems Engineering Research Center researching ways to address challenges
  - Beginning with space systems (COSATMO models)
  - Extendable to other DoD domains
- Workshop objectives
  - Understand, prioritize ground system cost estimation needs, opportunities
  - Identify sources of expertise, data

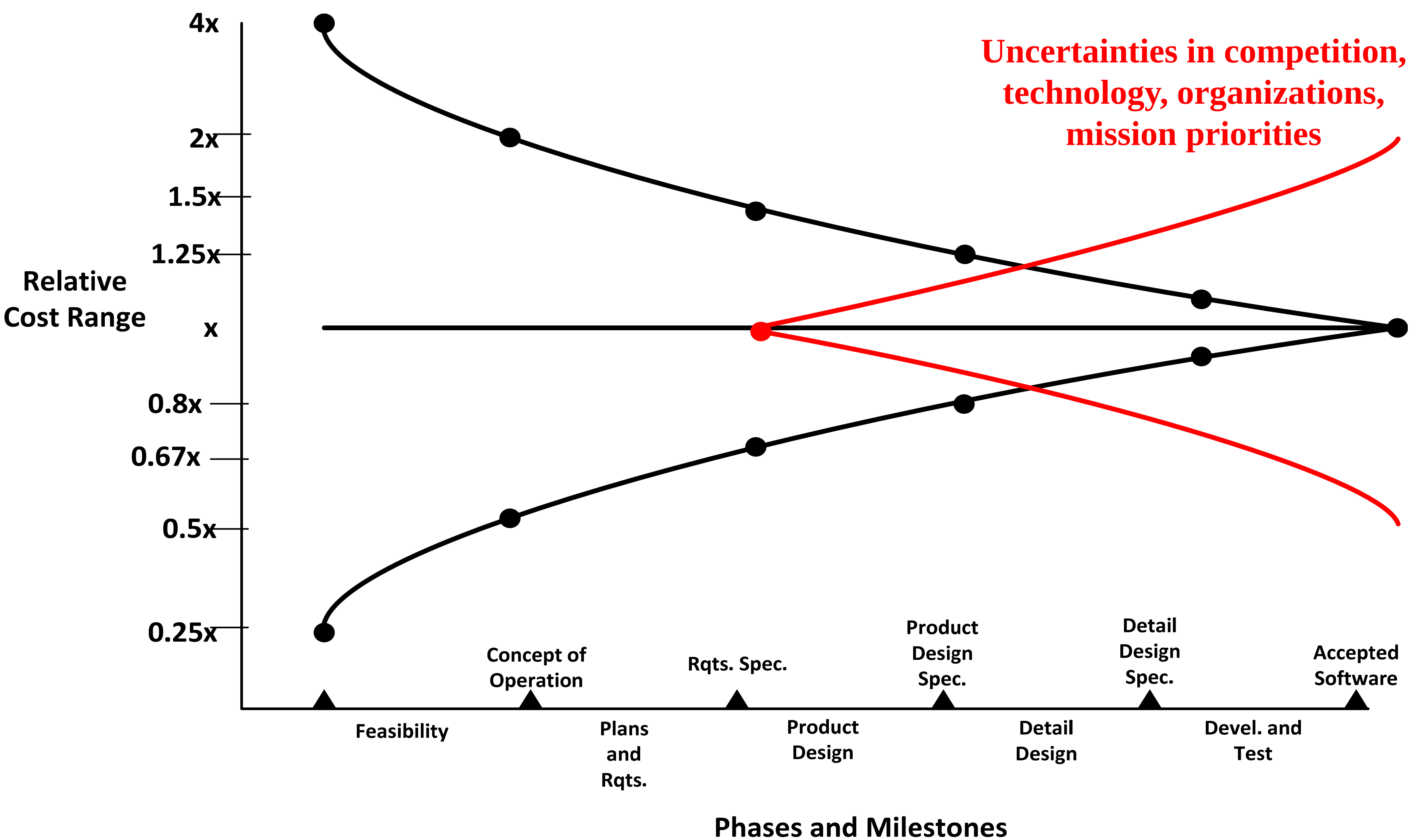
Software Estimation: The Receding Horizon



Current and Future Estimation Challenges

- Emergent requirements
  - Cannot prespecify requirements, cost, schedule, EVMS
  - Need to estimate and track early concurrent engineering
- Rapid change
  - Long acquisition cycles breed obsolescence
  - Need better models for incremental development
- Net-centric systems of systems
  - Incomplete visibility and control of elements
- Model, COTS, service-based, Brownfield systems
  - New phenomenology, counting rules
- Major concerns with affordability
  - Multi-mission ground system challenges

Rapid Change Creates a Late Cone of Uncertainty
– Need evolutionary/incremental vs. one-shot development

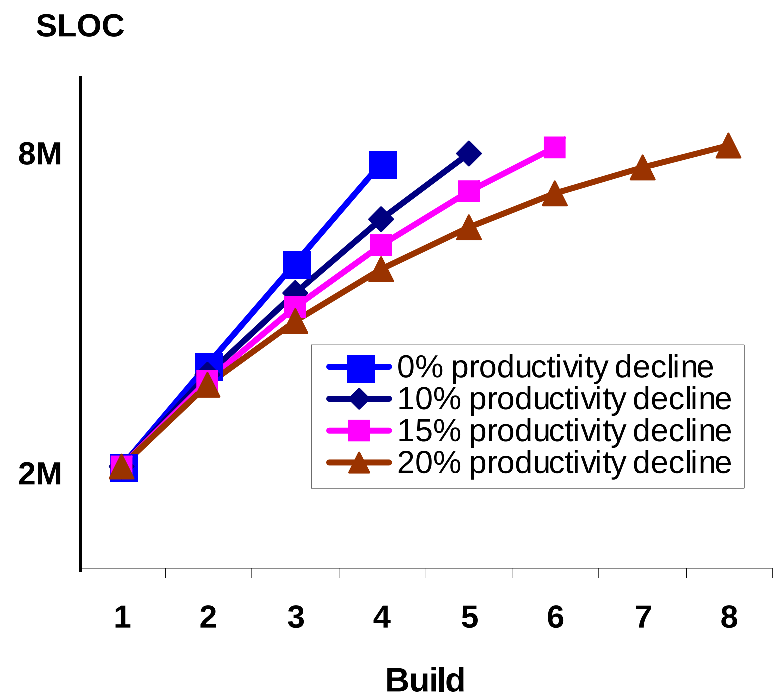


Incremental Development Productivity Decline (IDPD)

- Example: Site Defense BMD Software
  - 5 builds, 7 years, $100M; operational and support software
  - Build 1 productivity over 300 LOC/person month
  - Build 5 productivity under 150 LOC/PM
    - Including Build 1-4 breakage, integration, rework
    - 318% change in requirements across all builds
    - IDPD factor = 20% productivity decrease per build
  - Similar trends in later unprecedented systems
  - Not unique to DoD: key source of Windows Vista delays
- Maintenance of full non-COTS SLOC, not ESLOC
  - Build 1: 200 KSLOC new; 200K reused@20% = 240K ESLOC
  - Build 2: 400 KSLOC of Build 1 software to maintain, integrate

Effects of IDPD on Number of Increments

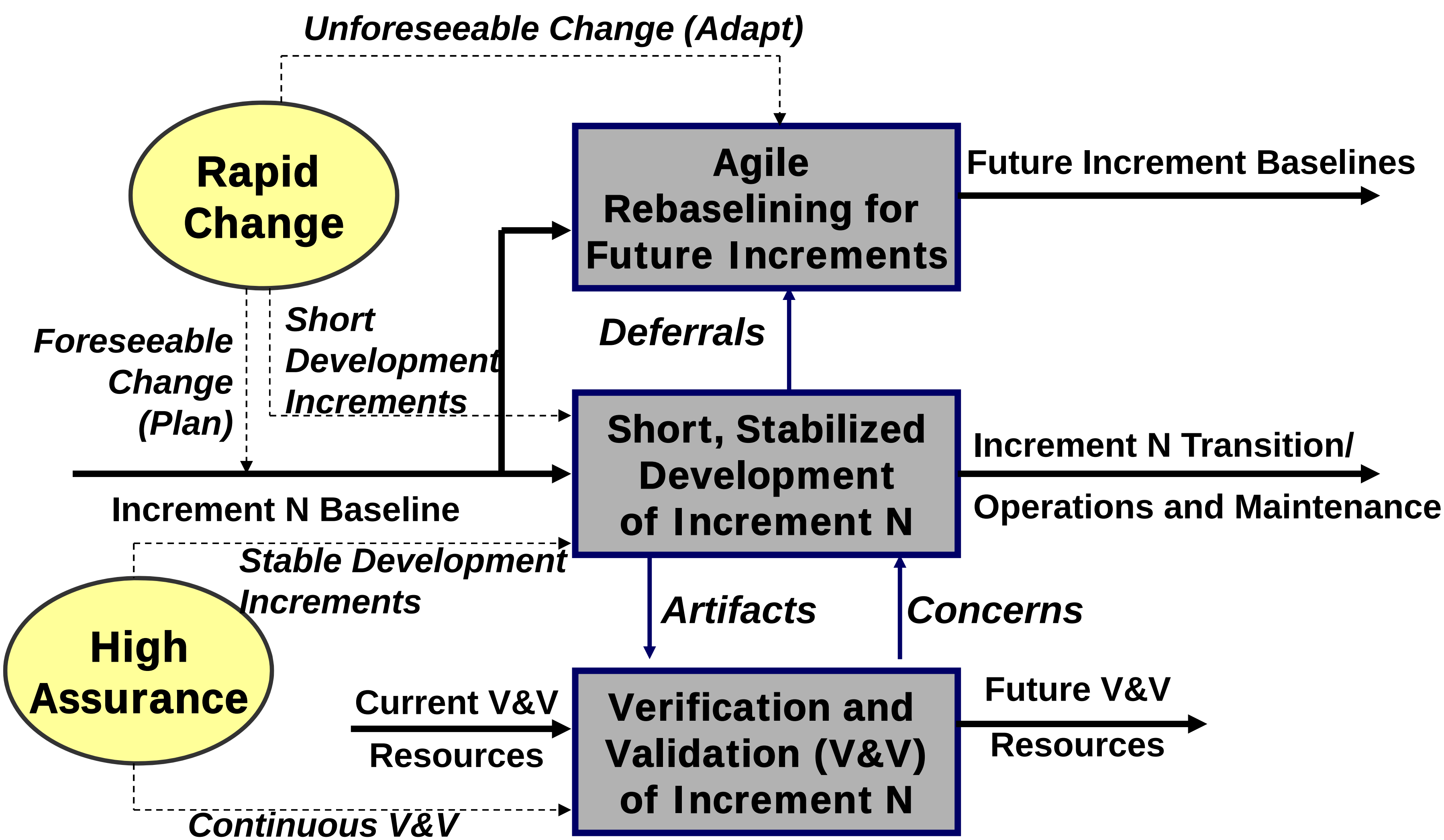
- Model relating productivity decline to number of builds needed to reach 8M SLOC Full Operational Capability
- Assumes Build 1 production of 2M SLOC @ 100 SLOC/PM
  - 20000 PM/ 24 mo. = 833 developers
  - Constant staff size for all builds
- Analysis varies the productivity decline per build
  - Extremely important to determine the incremental development productivity decline (IDPD) factor per build



Multi-Mission Ground Systems Costing

- **Product Line Engineering**
  - Identify multi-mission commonalities and variabilities
  - Identify fully, partially sharable commonalities
  - Develop plug-compatible interfaces for variabilities
- **Product Line Costing (COPLIMO) Parameters**
  - Fractions of system fully reusable, partially reusable and cost of developing them for reuse
  - Fraction of system variabilities and cost of development
  - System lifetime and rates of change
- **Product Line Life Cycle Challenges**
  - Layered services vs. functional hierarchy
  - Modularization around sources of change
  - Version control, CTS refresh, and change prioritization
  - Balancing agilty, assurance, and affordability

Risk-Driven Scalable Spiral Model: Increment View



Summary

- **Current and future trends create challenges for ground system cost estimation**
  - **Mission challenges: emergent requirements, rapid change, net-centric systems of systems, COTS, clouds, apps, widgets, high assurance with agility, multi-mission systems**
- **DoD Systems Engineering Research Center researching ways to address challenges**
  - **Beginning with space systems (COSATMO models)**
  - **Extendable to other DoD domains**
- **Workshop objectives**
  - **Understand, prioritize ground system cost estimation needs, opportunities**
  - **Identify sources of expertise, data**

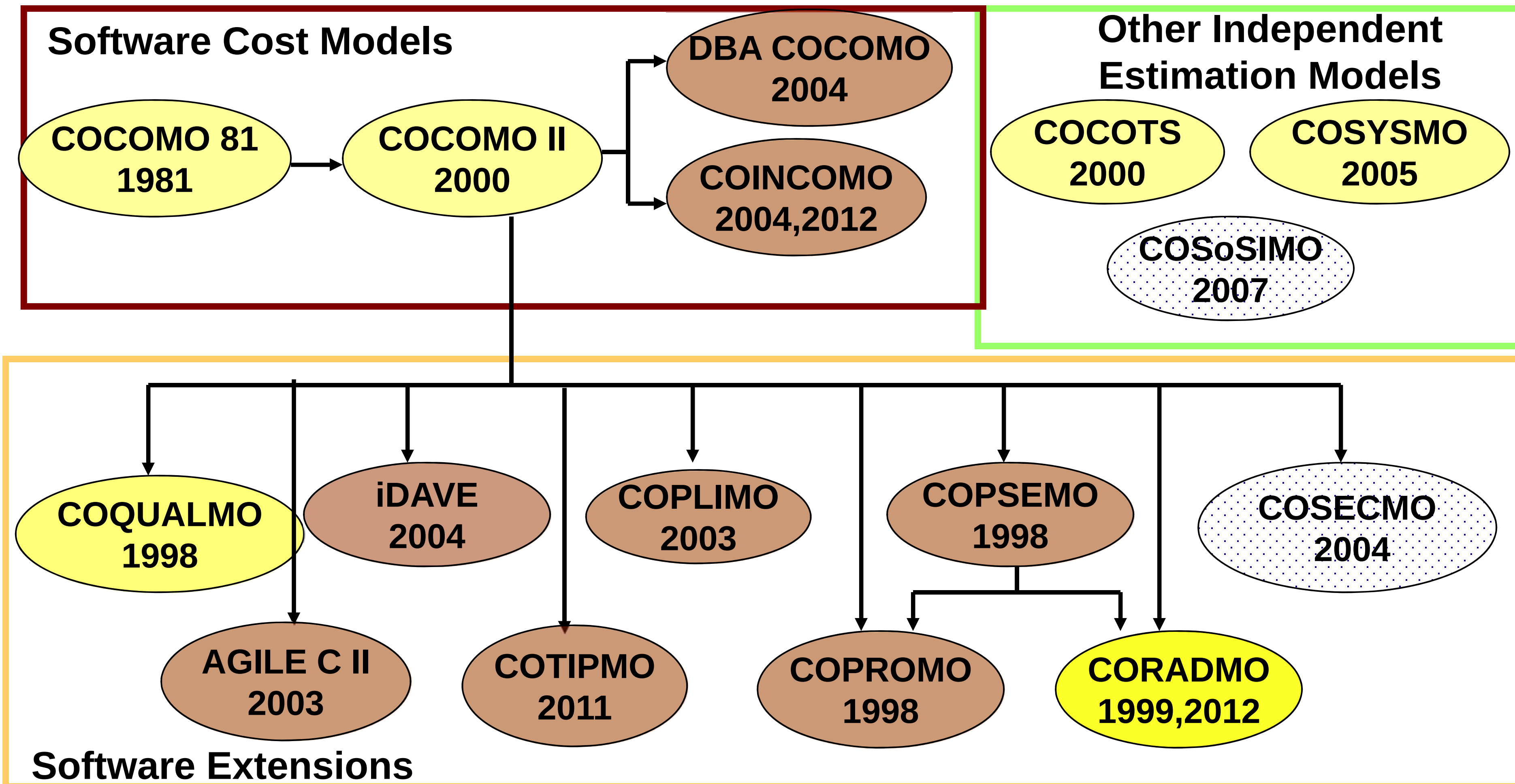
COSATMO Concept

- Focused on current and future satellite systems
  - Accommodating rapid change, evolutionary development, Net-Centric SoSs, Families of systems, DI2E SWASe's
    - Software, Widgets, Assets, Services, etc.
  - Recognizes new draft DoDI 5000.02 process models
    - Hardware-intensive, DoD-unique SW-intensive, Incremental SW-intensive, Accelerated acquisition, 2 Hybrids (HW-, SW-dominant)
  - Supports affordability analyses (total cost of ownership):
    - Covers full life cycle: definition, development, production, operations, support, phaseout
    - Covers full system: satellite(s), ground systems, launch
    - Covers hardware, software, personnel costs
- Extensions to cover systems of systems, families of systems
- Several PhD dissertations involved (as with COSYSMO)
  - Incrementally developed based on priority, data availability

COSATMO Tentative Model

- **Total satellite system cost =**
  - **System engineering cost**
  - **+ Satellite software cost**
  - **+ Satellite vehicle hardware development and production cost**
  - **+ Launch cost**
  - **+ Initial ground software cost**
  - **+ Initial ground facility cost**
  - **+ Operation & support cost**
- **Model as sum of submodels is new structure in COCOMO family**

COCOMO Family of Cost Models



Legend:

Model has been calibrated with historical project data and expert (Delphi) data

Model is derived from COCOMO II

Model has been calibrated with expert (Delphi) data

Dates indicate the time that the first paper was published for the model

COSATMO Submodel Starting Points

- System engineering: COSYSMO, perhaps with add-ons
- Satellite vehicle hardware development and production: Current Aerospace hardware cost model(s); exploring extensions of COSYSMO for hardware cost estimation
- Satellite vehicle, ground system software development: COCOMO II, COCOTS, perhaps with add-ons
- Launch model: similarity model, based on vehicle mass, size, orbit
- Ground system equipment, supplies: construction, unit-cost, services cost models
- Operation & support: labor-grade-based cost models, software maintenance models

My Tentative Research Objectives

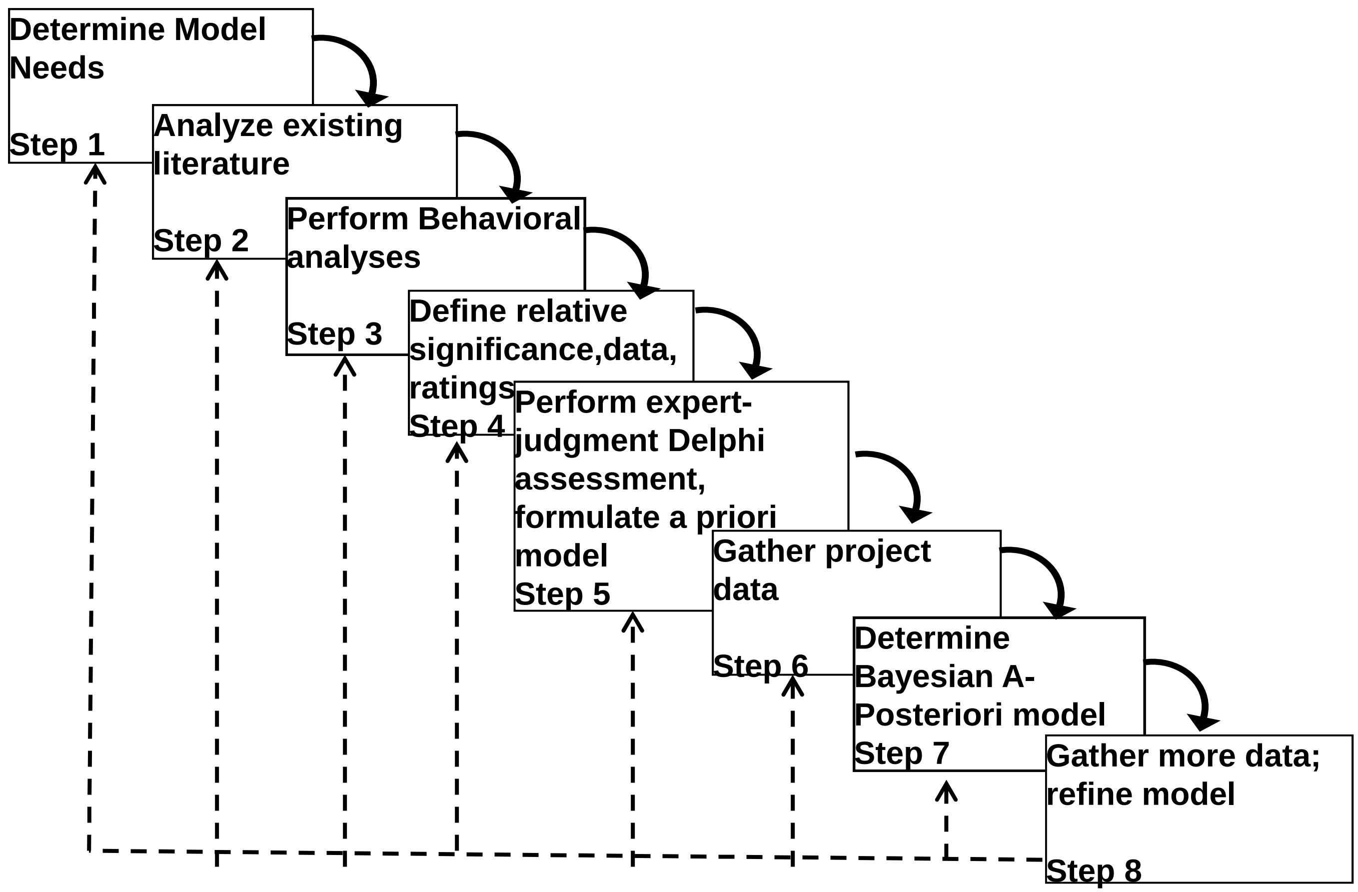
- **Provide improved cost estimation capabilities for the portions of and changing needs of space systems that are most needed and most currently tractable, including availability of calibration data. For example, SMC's main current concern is better estimation of post-deployment operations and sustainment costs.**

- **Develop a framework of cost estimation methods best suited for the various aspects of current and future space systems and other domains, such as the use of unit costing for production, acquisition, and consumables costs, and the use of activity-based costing for operations and sustainment labor costs.**

- **Prioritize the backlog of estimation models to be developed next.**

USC-CSSE Modeling Methodology

- concurrency and feedback implied



Survey Results: Key COSATMO Cost Drivers
COCOMO-SSCM Forum, October 24, 2013

- **Most Important**
  - **Complexity, Architecture Understanding, Mass, Payload TRL level, Technology Risk, and Requirements Understanding.**

- **Important**
  - **Reliability, Pointing Accuracy, Number of Deployables, Number of key sponsors, Data Rate, and Security Requirements for Communications.**

Upcoming COSATMO-Related Events

- **SERC project reviews March 19 prior to Conference on Systems Engineering Research (CSER 2014)**
  - **SERC reviews on USC campus**
  - **CSER at Redondo Beach Crowne Plaza, March 20-22**

- **USC-CSSE Annual Research Review April 29-May 1**
  - **Including COSATMO half-day workshop**
  - **On USC campus**

Summary

- Current and future trends create challenges for ground system cost estimation
  - Mission challenges: emergent requirements, rapid change, net-centric systems of systems, COTS, clouds, apps, widgets, high assurance with agility, multi-mission systems
- DoD Systems Engineering Research Center researching ways to address challenges
  - Beginning with space systems (COSATMO models)
  - Extendable to other DoD domains
- Workshop objectives
  - Understand, prioritize ground system cost estimation needs, opportunities
  - Identify sources of expertise, data

Backup Charts

Data definition topics for discussion

- Ways to treat data elements: Proposed DoD SRDR form updates
  - COTS, other OTS (open source; services; GOTS; reuse; legacy code)
  - Other size units (function points, object points, use case points, etc.)
  - Generated code: counting generator directives
  - Requirements volatility
  - Incremental development: productivity decline; deleted code
  - Rolling up CSCIs into systems
  - Cost model inputs and outputs (e.g., submitting estimate files)
- Scope issues
  - Cost drivers, Scale factors
  - Reuse parameters: Software Understanding , Programmer Unfamiliarity
  - Phases included: hardware-software integration; systems of systems integration, transition, maintenance
  - WBS elements and labor categories included
  - Parallel software WBS
  - Defect introduction and removal data

Incremental Development Data Challenges

- Breakage effects on previous increments
  - Modified, added, deleted SLOC: need Code Count with diff tool
- Accounting for breakage effort
  - Charged to current increment or I&T budget (IDPD)
    - IDPD effects may differ by type of software
  - "Breakage ESLOC" added to next increment
  - Hard to track phase and activity distributions
    - Hard to spread initial requirements and architecture effort
- Size and effort reporting
  - Often reported cumulatively
  - Subtracting previous increment size may miss deleted code
- Time-certain development
  - Which features completed? (Fully? Partly? Deferred?)

“Equivalent SLOC” Paradoxes

- Not a measure of software size
- Not a measure of software effort
- Not a measure of delivered software capability
- A quantity derived from software component sizes and reuse factors that helps estimate effort
- Once a product or increment is developed, its ESLOC loses its identity
  - Its size expands into full SLOC
  - Can apply reuse factors to this to determine an ESLOC quantity for the next increment
    - But this has no relation to the product’s size

Model, COTS, Service-Based, Brownfield Systems

New phenomenology, counting rules

- **Product generation from model directives**
  - Treat as very high level language: count directives
- **Sizing COTS and services use needs improvement**
  - Unrealistic to use COTS, services SLOC for sizing
  - Alternatives: function point elements, amount of glue code, activity-based assessment costing, tailoring parameters
- **Brownfield legacy constraints, re-engineering**
  - Re-engineer legacy code to fit new architecture
  - Apply reuse model for re-engineering
- **A common framework for reuse, incremental development, maintenance, legacy re-engineering?**
  - All involve reusing, modifying, deleting existing software

Summary of Recent Meetings

- 24 September at Aerospace
  - Presentations on satellite cost estimation
    - Notably, Lisa Colabella's survey of cost data gathering for Operations & Suppor (see backup chart)
- 24 October at COCOMO Forum
  - Started official COSATMO modeling effort
  - Got 1st draft of most important cost drivers, list of experts
- 18 November at JPL
  - Presentations on their satellite cost models, including some operations modeling
- 18 December at SMC
  - Obtained pointers to some of their operation & support data

Extension to Other Domains

- Will the tentative COSATMO model be a suitable model for other domains?
- Total system cost =

  System engineering cost
  + Embedded software cost
  + Hardware development and production cost
  + Deployment cost
  + Initial logistics software cost
  + Initial logistics facility cost
  + Operation & support cost?