# Software Asset Management on System z

Mike Zelle
Tivoli WW IT Asset Management Marketing
SAM in SHARE Project Manager
mzelle@us.ibm.com

# Agenda

- Why Software Asset Management (SAM)
- The Discipline of Software Asset Management
- Benefits of Software Asset Management Activities
  - IBM Customer Successes
- How to Build a Software Asset Management Program
- How IBM Helps

# Why Software Asset Management?

- You may be paying for software you don't use
- You may be out of compliance
  - Normal IT operations can easily result in being out of compliance with software license terms
  - Can cause service disruptions to comply with license terms
  - Can cost significant $$ which take funds away from needed projects
- Your CEO & CFO are at risk of civil and/or criminal penalties if you do not have effective financial control procedures. (Sarbanes-Oxley Section 404)

# Software Asset Management – Business Benefits

**Compliance Risk Reduction**

- Software vendor license compliance audits
- World Wide impact of Sarbanes-Oxley Act Section 404
- US Office of Management and Budget Circular A-123

**Software Cost Management**

- Identification and reduction of no and low use software
- Efficient server consolidations
- Optimize software and hardware capacity upgrades
- Strong vendor contract negotiation leverage
- Reduced software fees through competitive replacement
- Improve operational performance and reduce maintenance costs
- Invoice validation
- Effective charge-back
- Disaster recovery and business continuation

**Opportunity Creation**

- Software dollars can be redeployed to fund new software acquisitions and projects

# The Discipline of Software Asset Management

**Software is Intellectual Property**

- Not like tangible assets (hardware)
- Use rights are 'licensed'
- Use rights are defined in the license agreement
  - What is a 'user'?

**Software Asset Management** (ITIL Definition)

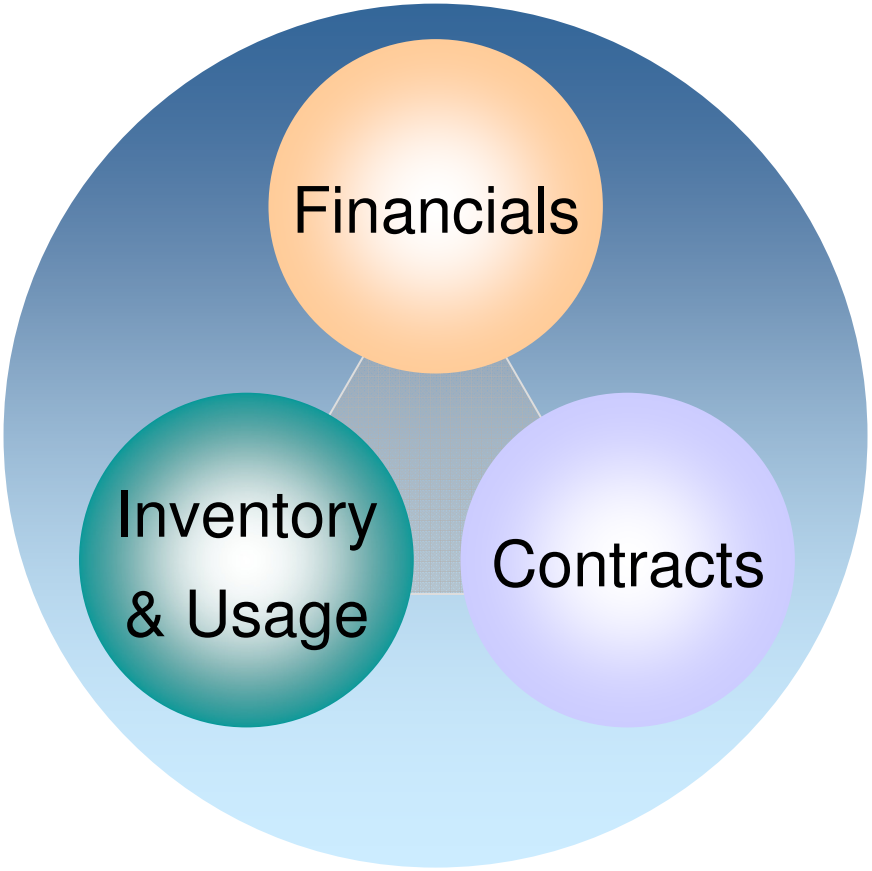
Software asset management (SAM) is the practice of **integrating people, processes and technology to allow software licenses and usage to be systematically tracked, evaluated and managed**. The goal of SAM is to reduce IT expenditures, human resource overhead and risks inherent in owning and managing software assets.

SAM includes maintaining software license compliance; tracking the inventory and usage of software assets; and maintaining standard policies and procedures surrounding the definition, deployment, configuration, use and retirement of software assets. SAM represents the software component of IT asset management, which also includes hardware asset management (to which SAM is intrinsicly linked by the concept that without effective inventory hardware controls, efforts to control the software thereon will be significantly inhibited).

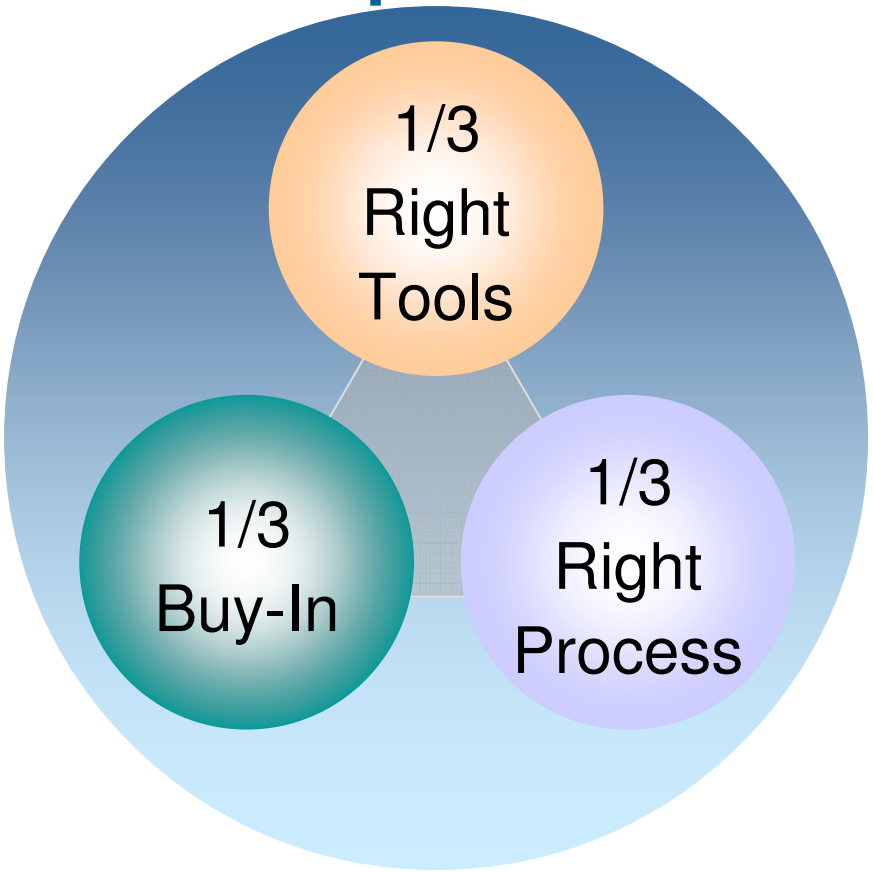
ITIL Definition – Wikipedia 2008

# Software Asset Management Sustaining Success

## Gartner



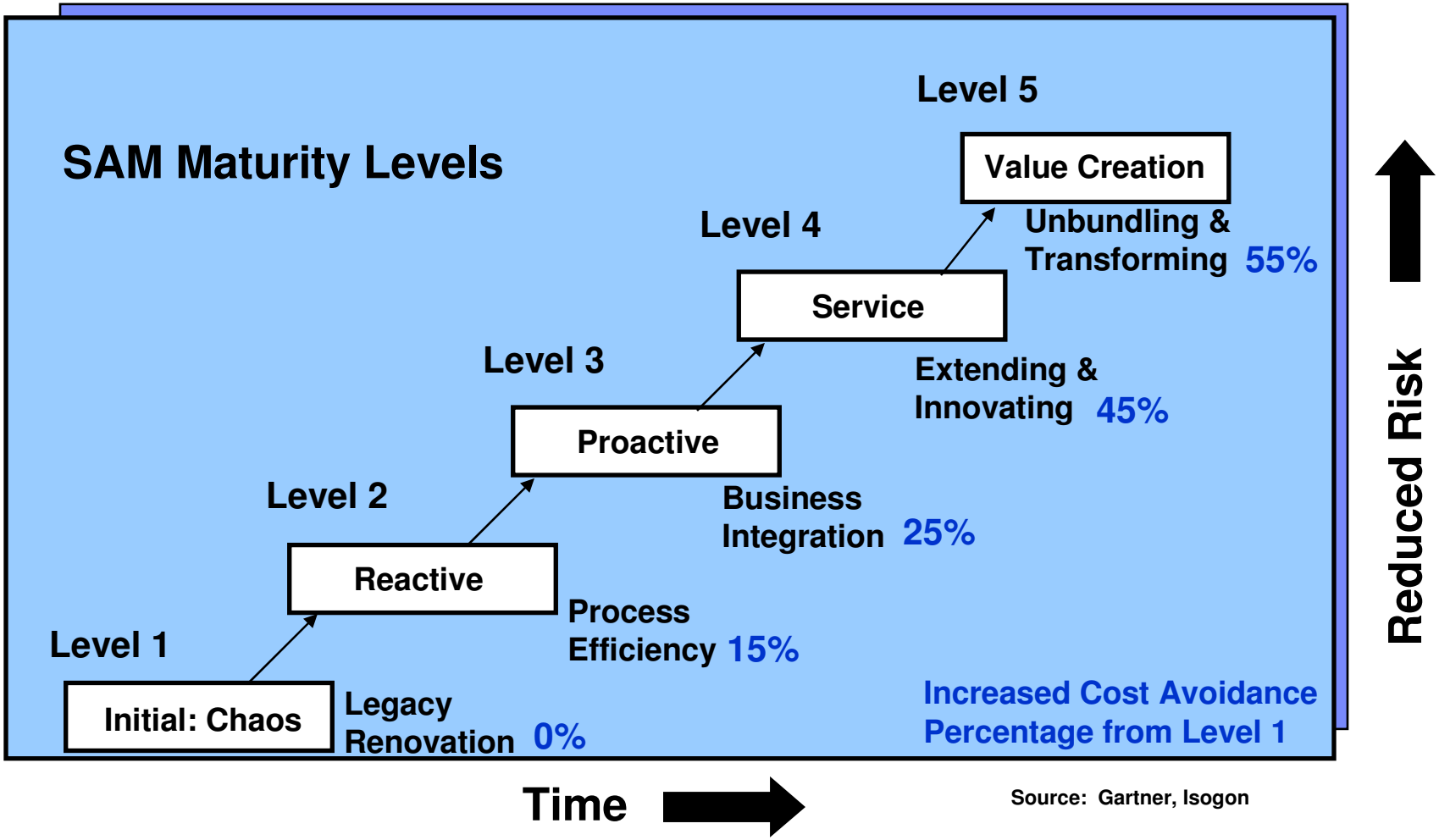

## IBM Customer Experience



# Why Are Some Companies Less Successful?

- No executive buy-in
- Little or no expertise in SAM processes
- Inability to justify the resources required
- Internal conflicts between the groups managing the technical environments, budgets, contracts and business needs associated with software
- No way to pull it all together
- Little awareness of the potential $$$ savings
- Little understanding of where they are today

# Software Asset Maturity Maturity Levels



# Benefits from SAM Activities

I have it

I use it

No contract for it

✓Verify license compliance

I have it

I use it

I'm contracted for it

✓Invoice validation, leverage contract negotiations

I don't have it

I don't use it

I'm contracted for it

✓Cancel maintenance or Verify install process

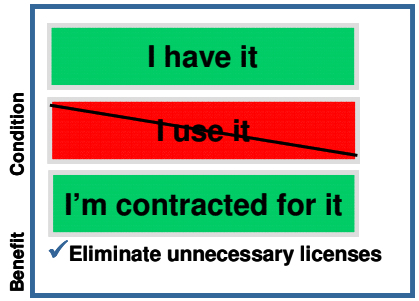
I have it

I don't use it

I'm contracted for it

✓Eliminate unnecessary licenses

# Success Stories



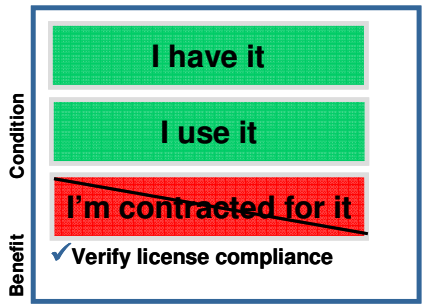

## Global Oil Company

- "In the first year, we identified 76 products that were seldom or never used and could be eliminated. We saved $1.25MM in software license and maintenance costs."
- "We had a modeling product that cost $144,000 per year but was used by only one application to produce a weekly service station sales report. Converting that application to a PC spreadsheet eliminated the service station sales report."
- "We also identified products that were licensed on multiple CPUs, but were only required on one. …we have been able to eliminate approximately 200 licenses."

## Forest Products Manufacturer

"There have been substantial savings over the years since we installed our mainframe auto-discovery tool. We were able to phase-out several low-use products and to eliminate their associated costs. We were able to drop several products based on research provided by the auto-discovery tool…"

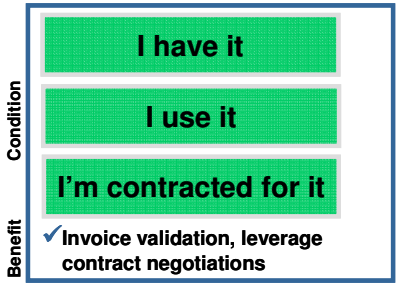
# Success Stories




**Financial Services Company**

"Over nine years ago, the organization set up a data repository containing information on contractual obligations, purchase dates, renewal dates and maintenance agreements related to mainframe software. We use a repository to monitor compliance with vendor contracts."

# Success Stories




**Insurance Company**

"Based on our knowledge of usage through our mainframe SAM product we are able to re-negotiate contracts with several of our largest vendors and maneuver the distribution of their products on our systems to become better aligned with our end users needs while also reducing costs."

**Cable Company**

"By using the reports generated by License Manager, we were able to view what software products our company was actually using, which allowed us to successfully renegotiate a greater than necessary software maintenance renewal." This saved us $1.8 million (CDN).

**Fortune 500 Manufacturer**

To give some insight to how effective the mainframe SAM product has been for us, consider our experience:

- "We started with 635 MIPS, 150 program products, and software expenses that were a little over $7,000,000; the software cost per MIP was around $11,000."
- "Four years latter we have ~ 1,100 MIPS, 65 program products, and software expenses around $4,400,000. Today our software cost per MIP is around $4,000."

# How to Build a SAM Program

- **Phase 1: Set the Groundwork**

  There are, of course, several premises and assumptions that are typically in place to support this work. If the items below are not present already, they become the first steps taken by the SAM team.

  - Software inventory and usage information is identified, maintained, and stored in a repository.
  - Contractual/licensing information is, at a minimum, entered and maintained in a spreadsheet and includes renewal dates, costs, products, function, and platforms covered.
  - The SAM team is responsible for all platforms and has determined which platform to focus on. Typically, the initial focus is on the mainframe for the greatest cost savings.

- **Phase 2: Identify Products Up for Renewal**

  After setting the groundwork, the SAM team identifies the contracts with associated products that are up for renewal within a year. To do this, the team:

  - Organizes contract renewals by dollar value by quarter due.
  - Develops a continuous negotiation strategy for vendors who represent over 20% of the software budget.
  - Addresses the next quarter renewals that meet a minimum dollar impact. For example, if a contract is up for renewal in 60 days and the license only costs $5,000, the SAM team may focus on other higher- priced contracts.
  - This approach will leave time for proactive negotiations and advanced preparations for the situations that develop throughout the year.

# How to Build a SAM Program

- **Phase 3: Identify Products with No or Low Usage**

  Next, the SAM team identifies software products with no usage and those products with low usage over a period of at least three months. Establishing low usage can be tricky. To do this effectively, the team needs to review the product's functionality along with the usage data analysis, rather than simply use criteria such as a fixed number of product executions as a threshold. The team will:

  - Evaluate the product functionality. The usage reported may or may not be consistent with the functionality. For example, security products and some operating system components are only executed once during start-up.
  - Review the frequency and time frame when the product was used. The usage reported may or may not be consistent with the product's expected processing frequency. For example, applications may be scheduled to run on a weekly, monthly, quarterly, or annual basis.
  - It's also important to determine the location of products and to know the users associated with all products. This is particularly true of no and low usage software products.

- **Phase 4: Eliminate Unnecessary Products**

  The SAM team is now in a position to use this initial set of information effectively.

  - This begins by eliminating products that are not used after gaining approval of the users. With that approval, the SAM team can contact the associated vendor to renegotiate and achieve direct cost savings.
  - In a similar fashion the SAM team can then move on to contact vendors and renegotiate maintenance contracts for those software products with low usage.
  - Understanding the location of low usage products provides an incentive to work with the capacity planning group to reposition low usage products to smaller CPUs that have lower cost metrics. With this information, the team can adjust the physical organization of products to achieve direct cost savings and avoid costs by deferring or delaying hardware additions and upgrades.
  - The software inventory and usage information enables the SAM team to pursue another cost saving initiative by identifying and potentially eliminating redundant software products. For example, the team may uncover multiple security applications or testing aids. The usage data along with the cost will identify whether there are clear opportunities to pursue. The team can then interact with users and conduct competitive replacement discussions with the associated vendors to turn the potential into actual cost savings.

# How to Build a SAM Program

- **Phase 5: Eliminate Multiple Versions of Products**

  In a similar fashion, the team can evaluate and eliminate multiple versions of software products. For example, there may be multiple versions of a database used by the applications development team. The combination of cost, usage, and user data may clarify opportunities to adjust the product mix as necessary to achieve cost savings. By using the contractual and licensing information in conjunction with the inventory and usage data, you will find a number of cost savings opportunities. The SAM team can:

  - Identify products that were purchased but never installed. Armed with this information, the team can contact users to determine their continuing interest in the product and then potentially contact vendors and renegotiate.
  - Consolidate contracts based on platform, location, and type of agreements. This information can be used to achieve cost savings. For example, the company may:
    1. Have separate contracts for mainframe, UNIX, and Windows with one vendor. These could be combined into one Enterprise agreement.
    2. Have a contract for a software product to be used in two locations that could be combined into one Enterprise agreement.
    3. Have four CPU contracts for CPUs that reside in a single location. These could be combined into one Site agreement for that location.
    4. Ensure software compliance in order to avoid costly penalties.

  The contractual and licensing information can also be used to reconcile vendor invoices.

- **Phase 6: Resolve Discrepancies**

  There are often significant opportunities for error in the vendor's information and billing process. To guard against errors, validate year-to-date invoices against inventory and license terms and identify excess inventory and fees consistent with license terms.

  In a similar manner, the SAM team can compare license metrics to the installed metrics. For example:

  - A mainframe software product may be licensed for 2,000 MIPS on a CPU, but may only be using 500 MIPS.
  - A distributed software application may be licensed for 200 seats, but in reality, only 150 seats are being used.

  The team can work with the vendor to resolve the discrepancies and adjust to achieve cost savings.

# How to Build a SAM Program

- **Phase 7: Generate Long-Term Cost Savings**

  The SAM team can set up a virtual license warehouse to help generate long-term cost savings and efficiencies. For example, the company may have 5,000 Microsoft Office licenses, but only 4,500 are deployed. This leaves 500 Microsoft Office licenses in the virtual license warehouse. Then, when a new employee is hired, rather than buying an additional Microsoft Office license, the license can be taken from the warehouse.

  On an ongoing basis, the SAM team can initiate formal tracking of SAM events to log all major activities that occur throughout the year. This information can form the basis for:

  - Documenting accomplishments throughout the year
  - Justifying additional resources (staff, software, etc.)
  - Summarizing cost savings and cost avoidance
  - Listing new acquisitions
  - Highlighting process and performance improvements.

# Only IBM delivers Software Asset Management for both distributed and mainframe environments

What does each software asset cost now and in the future?

Over Licensed? (Potential Savings)
Under Licensed? (Compliance Risk)

What software assets are in use?

What software do I have?

**Contracts & Financials**

**License Optimization**

**License Use**
**Software Use**

**Inventory**

**IBM Tivoli Asset Management for IT**

| IBM Tivoli | IBM Tivoli |
|---|---|
| License Compliance Manager | License Compliance Manager for z/OS |

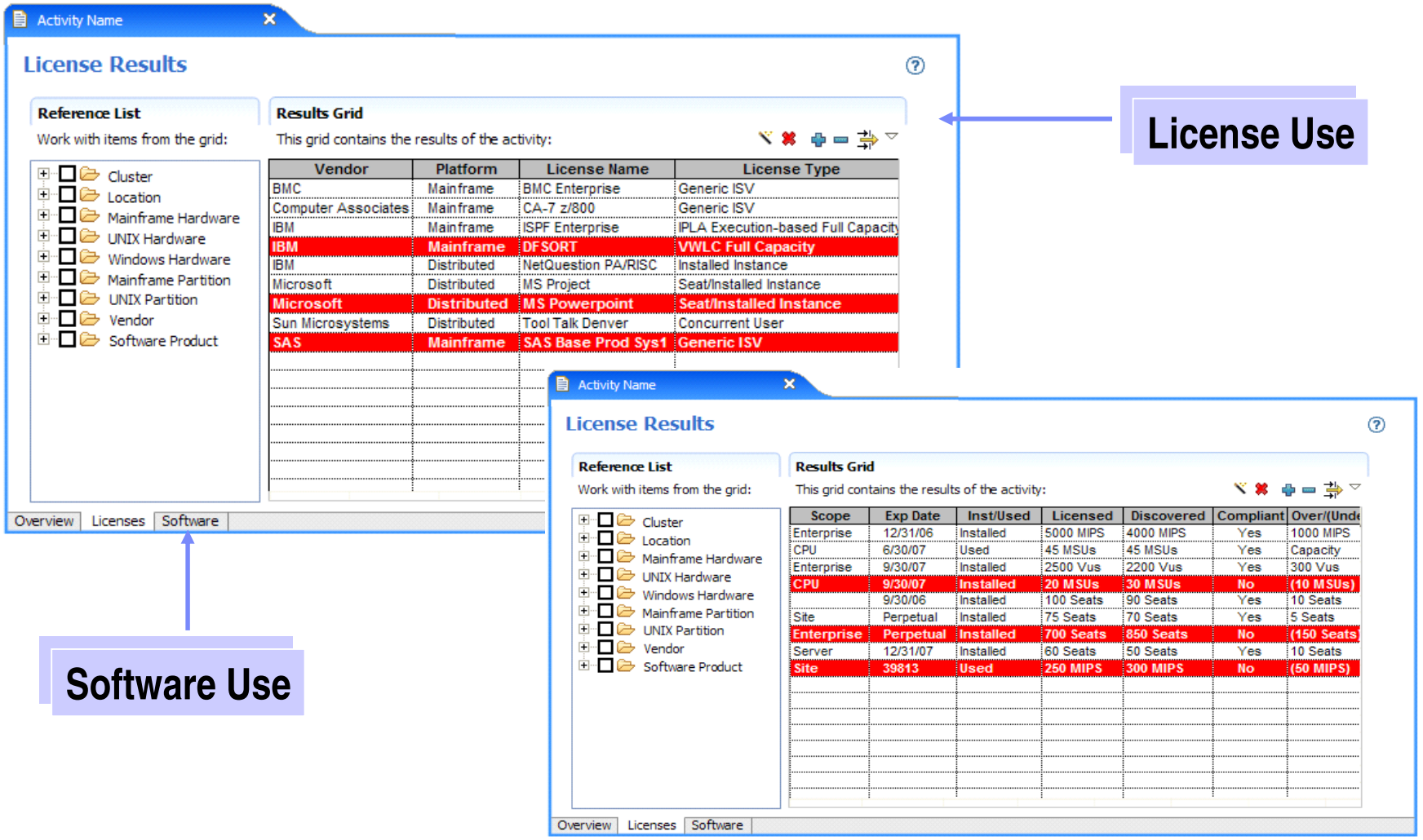
PCs

Servers

System z

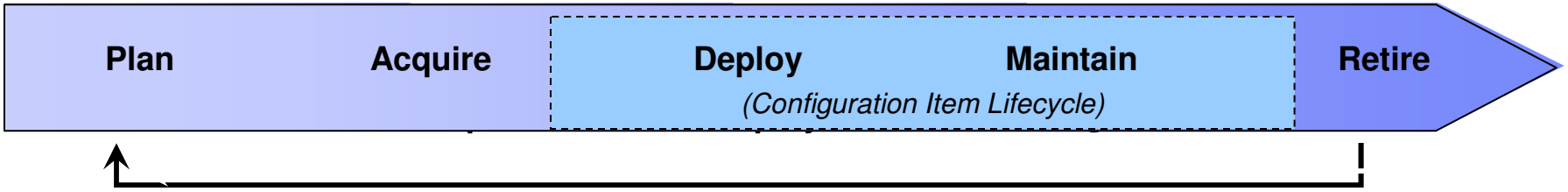
**All four questions must be answered to align software spending with business goals**

# Tivoli License Compliance Managers Sample GUI



# Tivoli Asset Management for IT

*Visibility and Control over Assets and their Impact to the Business*




**IT Assets in Operational or Productive Use**

- Align IT to corporate strategy
- Plan technology for new initiatives
- Plan technology refresh
- Plan for asset purchase or lease
- Negotiate vendor contracts
- Check inventory and plan for asset reuse
- Determine asset reliability
- Support IT budgeting

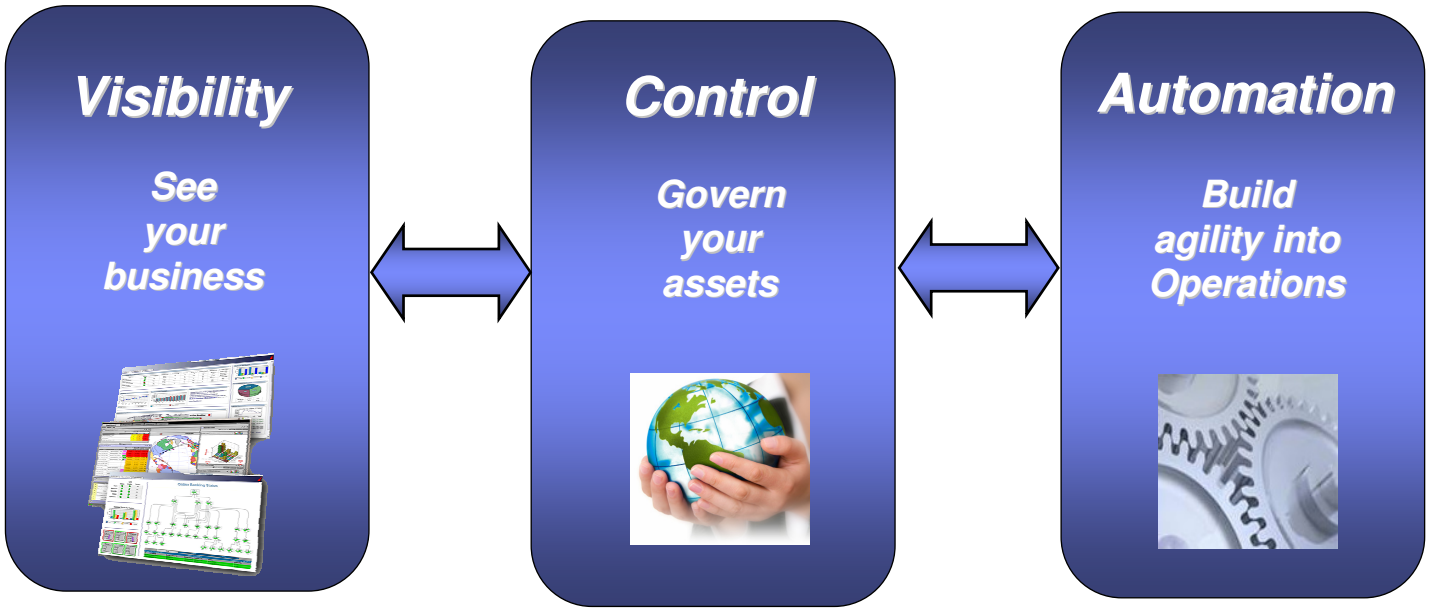
- Negotiate agreements to maximize value
- Manage contracts with vendors
- Procurement
- Approvals of PRs and POs
- Receive assets
- Validate invoices
- Manage hardware leases
- Line of Business support

- Processes to ensure standardize committed ROI
- Creation of assets via procurement, inventory or vendor data
- Asset assignment
- Asset tracking
- Notification of asset receipt to end user
- Built-in tools to add asset attributes based on Asset Type
- Utilize Tivoli Service Request Manager to create deployment service tickets

- Implement support infrastructure and process to enhance productivity and satisfaction
- Standard Install, Move, Add, Change (IMAC)
- Asset reconciliation
- Risk assessment
- Software license compliance
- Govern changes & control configurations
- Track warranty and contract renewals

- Provide for orderly disposition of assets: disposed, auctioned, donated and employee purchase
- Manage end of life
- Track end of life options
- Adhere to regulatory requirements
- Manage disposed assets
- Provide finance with accurate end of life data

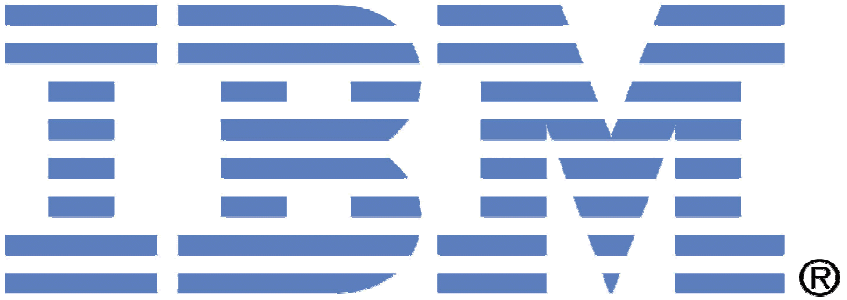
# IBM provides Asset & Service Management solutions that help clients realize their business goals through:




**Control Cost, Mitigate Compliance Risk, Improve Service Levels**

- ✓ See what you have, where it is, who is using it and what it costs
- ✓ Role based views
- ✓ Optimize software licenses, reduce over/under buying
- ✓ Reduce TCO of all strategic assets throughout their lifecycle
- ✓ Mitigate, regulatory, environmental and safety compliance risk
- ✓ Increase production and labor efficiency and spare parts optimization
- ✓ Reduce outages and failures with proactive asset management practices
- ✓ Automate workflow to enhance operational efficiencies
- ✓ Help IT and Operations make strategic purchasing decisions with reliable inventory data

- **Tivoli License Compliance Manager for z/OS**
  http://www-306.ibm.com/software/tivoli/products/license-compliance-mgr/
- **Tivoli License Compliance Manager (distributed systems)**
  http://www-306.ibm.com/software/tivoli/products/license-mgr/
- **Tivoli Asset Management for IT**
  http://www-306.ibm.com/software/tivoli/products/asset-management-it/



# What are the SAM Maturity Levels?

**Level 1: Chaos – Legacy Renovation**

The initial environment has ill-defined procedures, controls and a lack of adequate tools to track or manage assets resulting in a chaotic environment. The organization generally does not know what it owns, where it is located and who is using it. Unused hardware assets are often kept in a storage room with no control mechanisms or accountability by designated employees. Such enterprises often do not have a centralized purchasing or negotiation team. IT contracts are not tracked and are typically stored in folders in filing cabinets. There are no systems in place to track the intellectual capital and negotiation strategies associated with acquisition. Until the process is under statistical control, no orderly progress in process improvement is possible**.**

**Level 2: Reactive - Process Efficiency**

At Level 2, IS organizations create spreadsheets or a database to help track assets. An ITAM program is focused on counting the assets and involves annual physical inventories to identify & catalogue all PC equipment, and the process is now repeatable. The organization may use an auto-discovery tool to supplement this data. However, installation, move, add, changes processes are not consistently followed reducing the accuracy of the data. Reports are basic and lack sufficient detail to identify and resolve problems. Linkage and sharing of data with purchasing are sporadic at best. Often have multiple overlapping tools without centralized oversight or governance.

**Level 3: Proactive – Business Integration**

In Level 3, the process is well characterized and reasonably well understood. The organization moves into proactive management by defining its process in terms of software engineering standards and methods, and by making a series of organizational, methodological and SAM software tool improvements. Inventory data can now be linked with financial and contractual data to create a centralized view of how assets are performing. The repository and auto-discovery tools are integrated to provide real-time inventory and faster problem resolution.

Sources: Software Engineering Institute & Gartner 11-1-2001

# SAM Maturity Levels Continued…

**Level 4: Service – Extending & Innovating**

In Level 4, the SAM process is not only understood but it is quantified, measured and reasonably well controlled. The organization has metrics in place to measure the program's value and service levels can be created to meet broader business goals. Opportunities for savings are identified and communicated to the business units on a regular basis. Process and tools are used increasingly to control and enable faster procurement time frames, more efficient order tracking and improved invoice reconciliation. Inventory levels are managed to prevent overbuying while maintaining low-cost inventory stock levels and retiring or disposal of under performing assets. The organization is learning to project expected costs and problems with reasonable accuracy. This is when the most significant quality improvements begin.

**Level 5: Value Creation - Transformation**

At Level 5, organizations have not only achieved a high degree of control over their process, they have implemented and integrated all three tools of an ITAM program (repository, auto-discovery and software-usage) and have a major focus on improving and optimizing its operation. At this level, business units are charged back for computing services, TCO metrics are linked with ITAM metrics, and data from management and business applications are used to audit the efficiency and effectiveness of established business practices across all software assets. IT cost recovery encompasses usage based pricing models. The data on the process are used interactively to improve the process and achieve optimum performance.

Sources: Software Engineering Institute & Gartner 11-1-2001