

# The Good, The Bad and the Ugly

Sizing

Professional Services for early stage software product companies

Charles Rattray, Director, Professional Services - Tideway Systems Inc

# Abstract

- Complex software products fall within three extremes: being perfect; littered with coding bugs; and flawed by poorly implemented business logic. Developing a Services revenue model which remains effective, while the organization transitions towards producing near perfect software, and providing early customers value around their product purchase, can be a challenge. This presentation will outline the issues involved, discuss how this impacts the organization's success, and how to model and size the Services organization to maximize services revenue from a developing product.

# Agenda – A Case Study Approach

- The Challenge for Enterprise Software Product Buyers today
- An Example Software Product
- Working with the Product
- Resourcing the Deployment
- Sizing PS accordingly for Revenue growth
- Questions?

# Biggest Challenge for Enterprise Software Buyers Today?

- Answer:
  - The Vendor can get away with it!
    - Personal relationships – personal promises
    - Promise of direct product influence
    - What price value?
    - Consumer ready software – not expected
- But:
  - Not for much longer
    - SaaS
    - OpenSource
    - Business customers are consumers

# An Example Enterprise Software Product

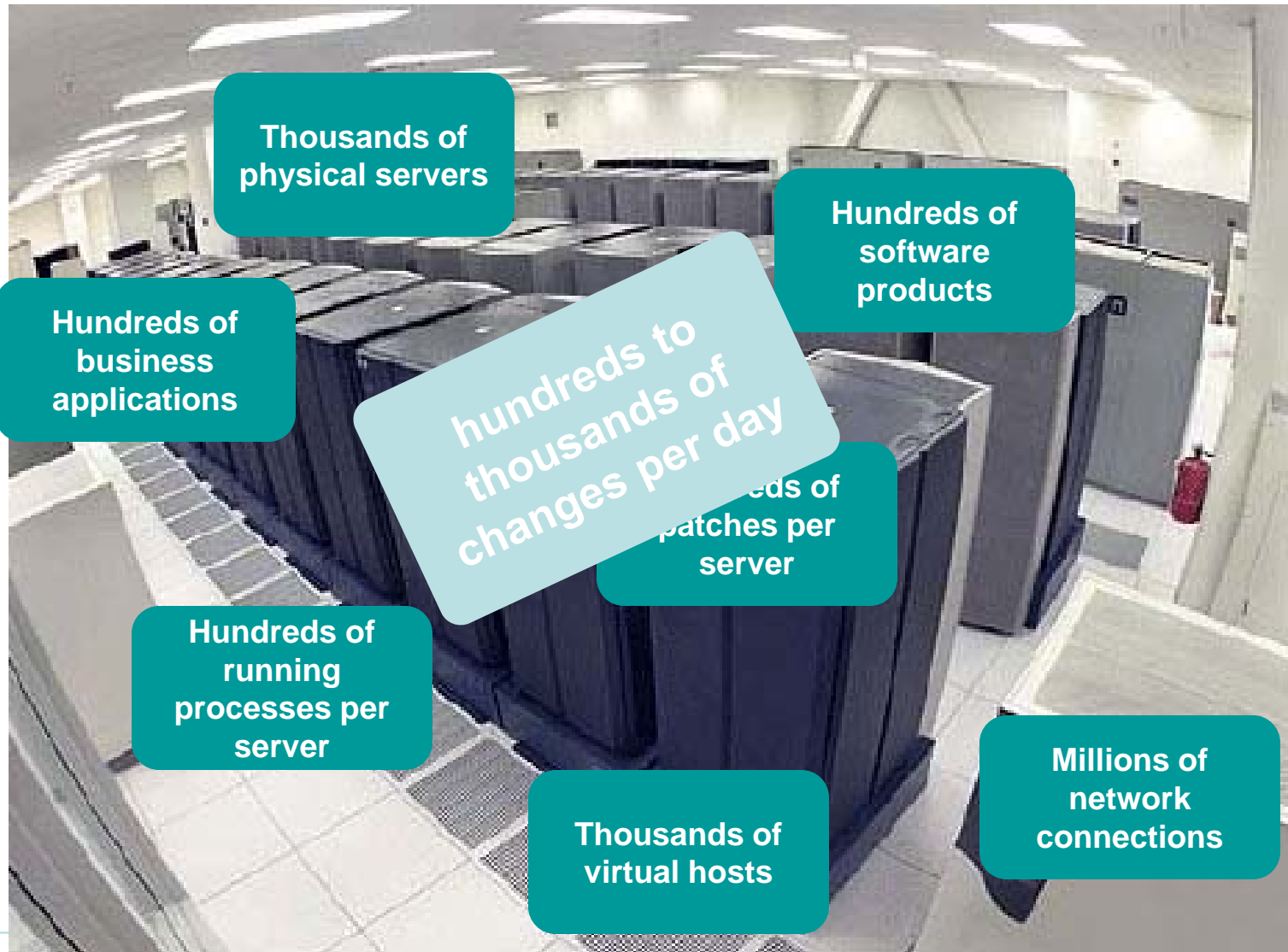
 Tideway Foundation

Application Dependency Mapping

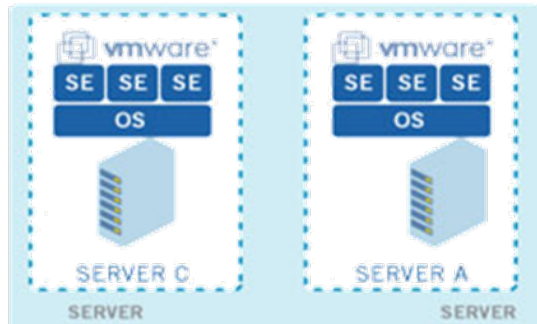
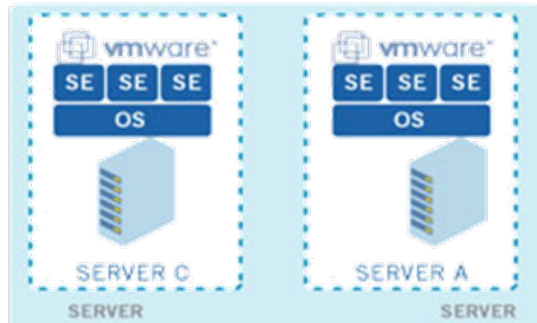
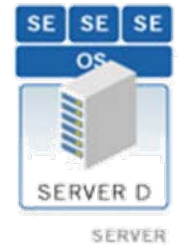
# What is Application Dependency Mapping?

- A usable definition:
  - ADM is about mapping business applications to your underlying physical and virtual infrastructure, providing a **360-degree view of how your infrastructure supports your business**. It connects disparate technology layers – from business applications to switches and all the dependencies in between – and provides the information in a single, automated view.

# A Reminder of the Challenge

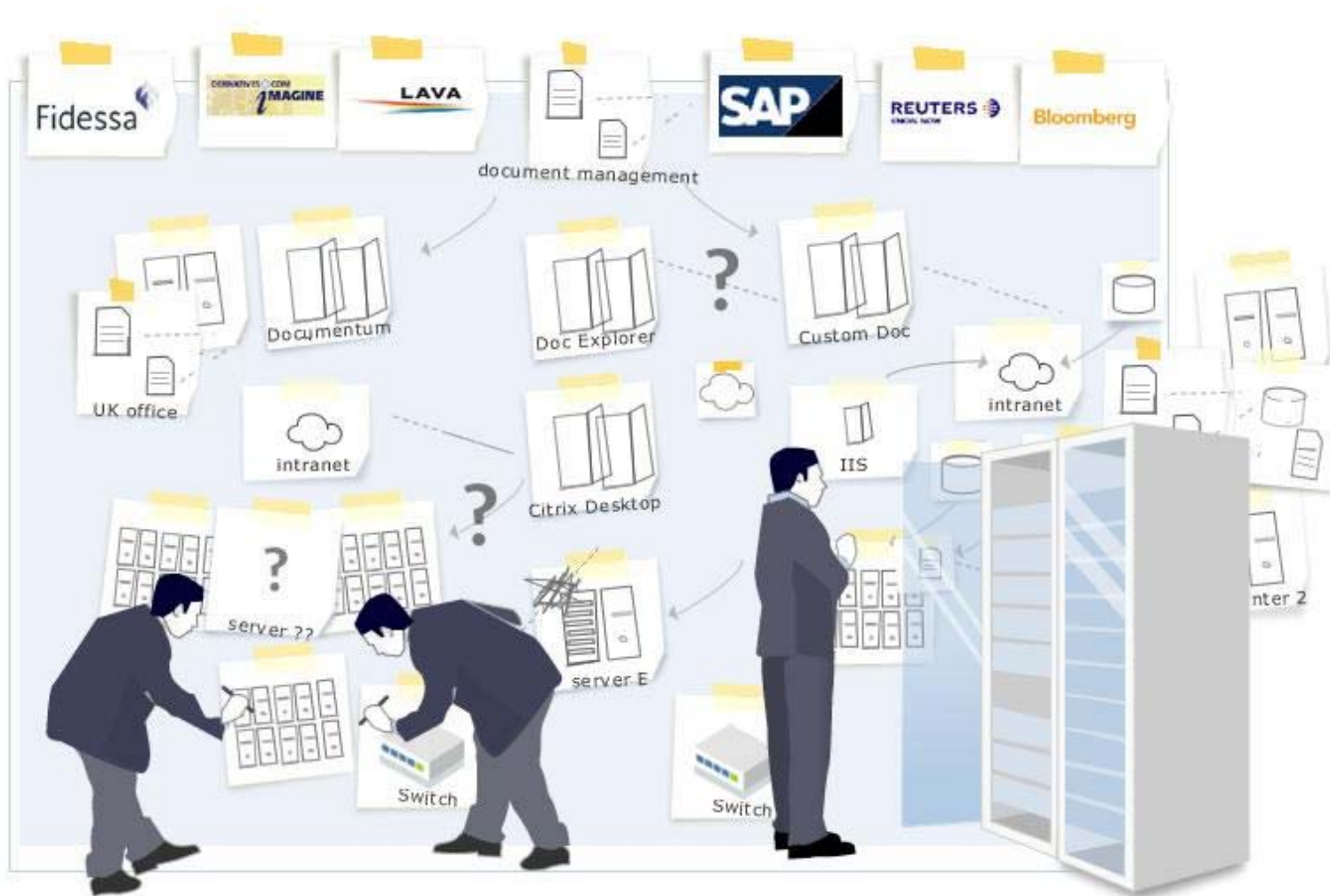


# IT Staff's View

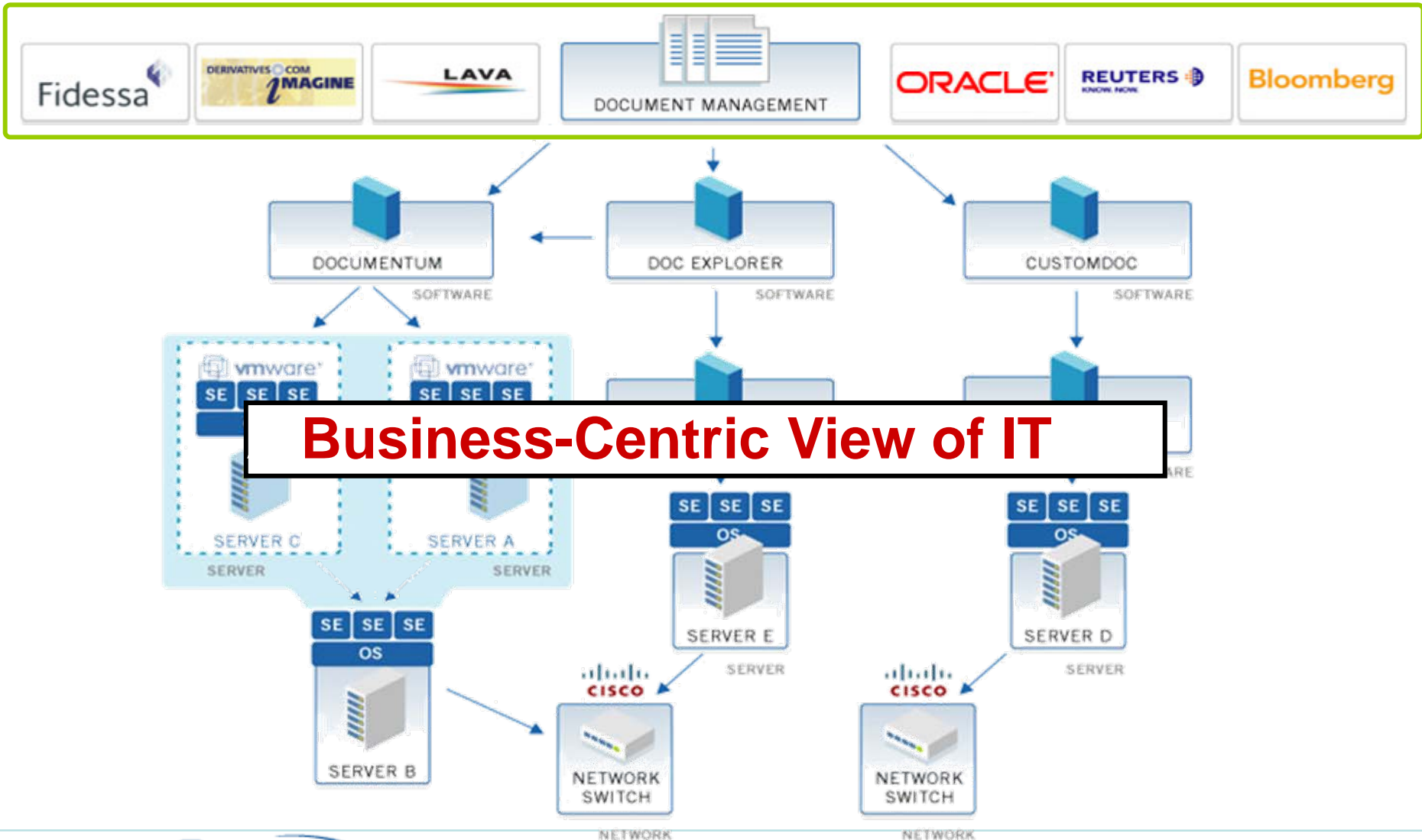




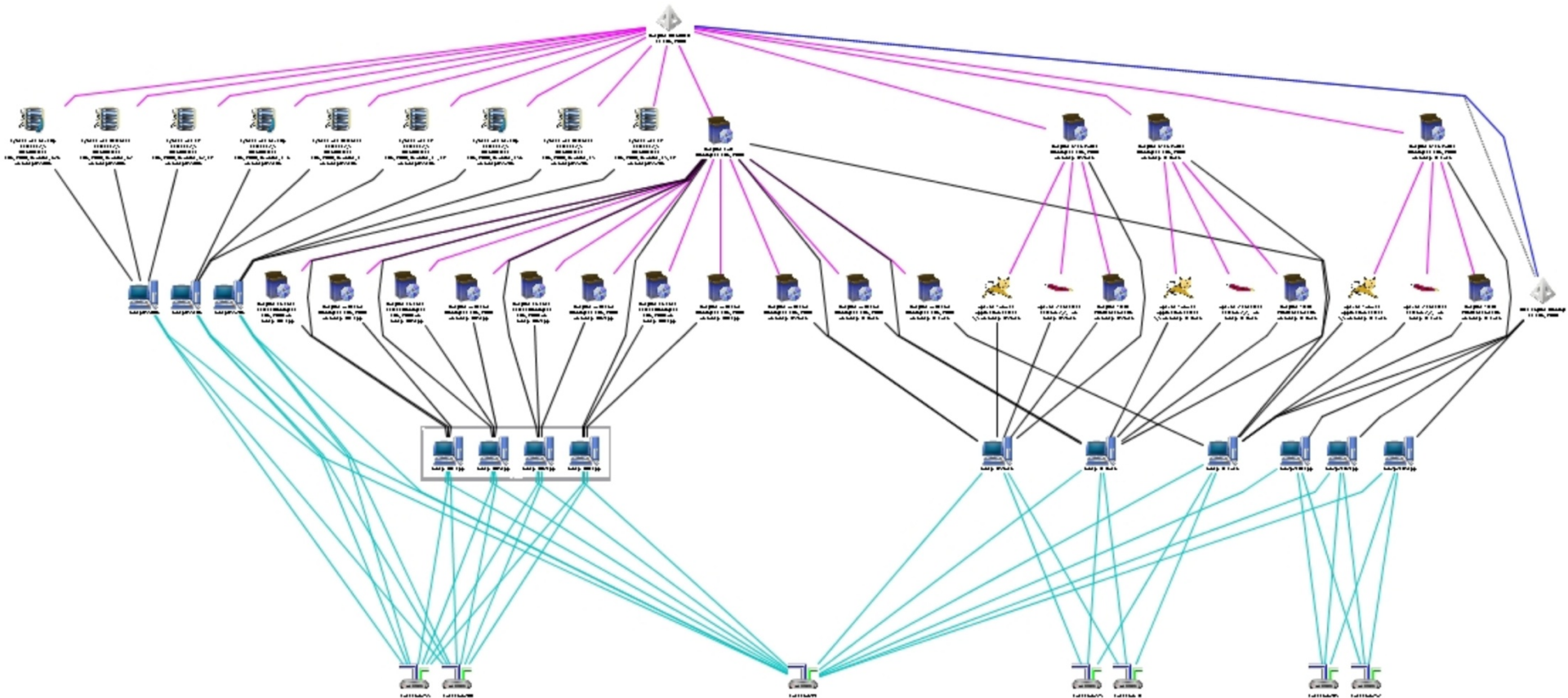
# Manual Approach Is Expensive and Inadequate



# Automation – Saves Cost / Reduces Risk

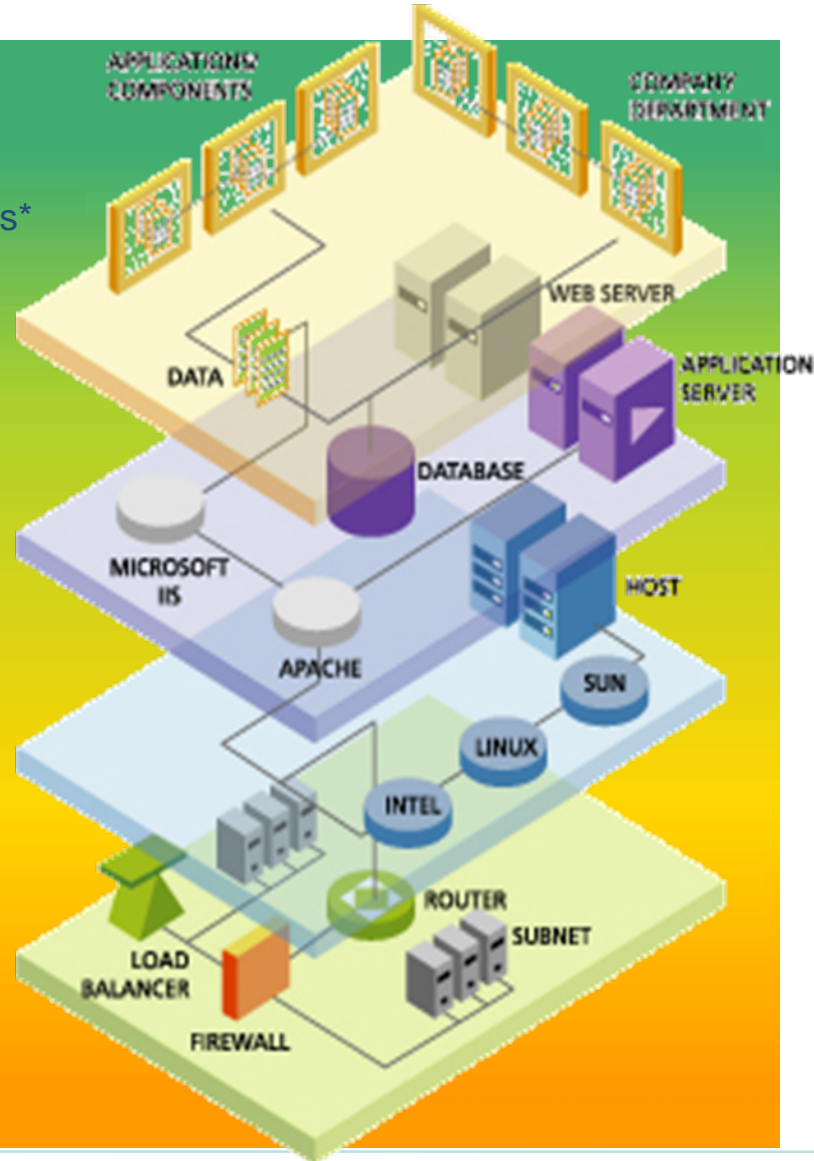


# The Solution - Application Dependency & Mapping



# Infrastructure Information

- |                        |   |                 |
|------------------------|---|-----------------|
| Full<br>Discovery      | <ul style="list-style-type: none"> <li>■ Description*</li> <li>■ Business owner*, support owner*, IT owner*</li> <li>■ Business continuity critical*, Disaster recovery status*</li> </ul>                  | <b>Business</b> |
|                        | <ul style="list-style-type: none"> <li>■ Documentation*</li> <li>■ Physical location*, room*</li> </ul>   |                 |
| Host<br>Information    | <ul style="list-style-type: none"> <li>■ Software product instance</li> <li>■ Software elements, interest level*</li> </ul>   | <b>Software</b> |
|                        | <ul style="list-style-type: none"> <li>■ Name</li> </ul>  | <b>Host</b>     |
| Host<br>Identification | <ul style="list-style-type: none"> <li>■ OS version</li> <li>■ Service pack, patch list</li> <li>■ Model #</li> <li>■ Serial #</li> <li>■ Number of processors and processor type</li> <li>■ RAM</li> </ul> |                 |
|                        | <ul style="list-style-type: none"> <li>■ IP address list</li> <li>■ Subnet, DNS domain</li> <li>■ NIS / Windows domain</li> <li>■ Network connection</li> </ul>   | <b>Network</b>  |
| Sweep<br>Scan          |   |                 |



# Great Product

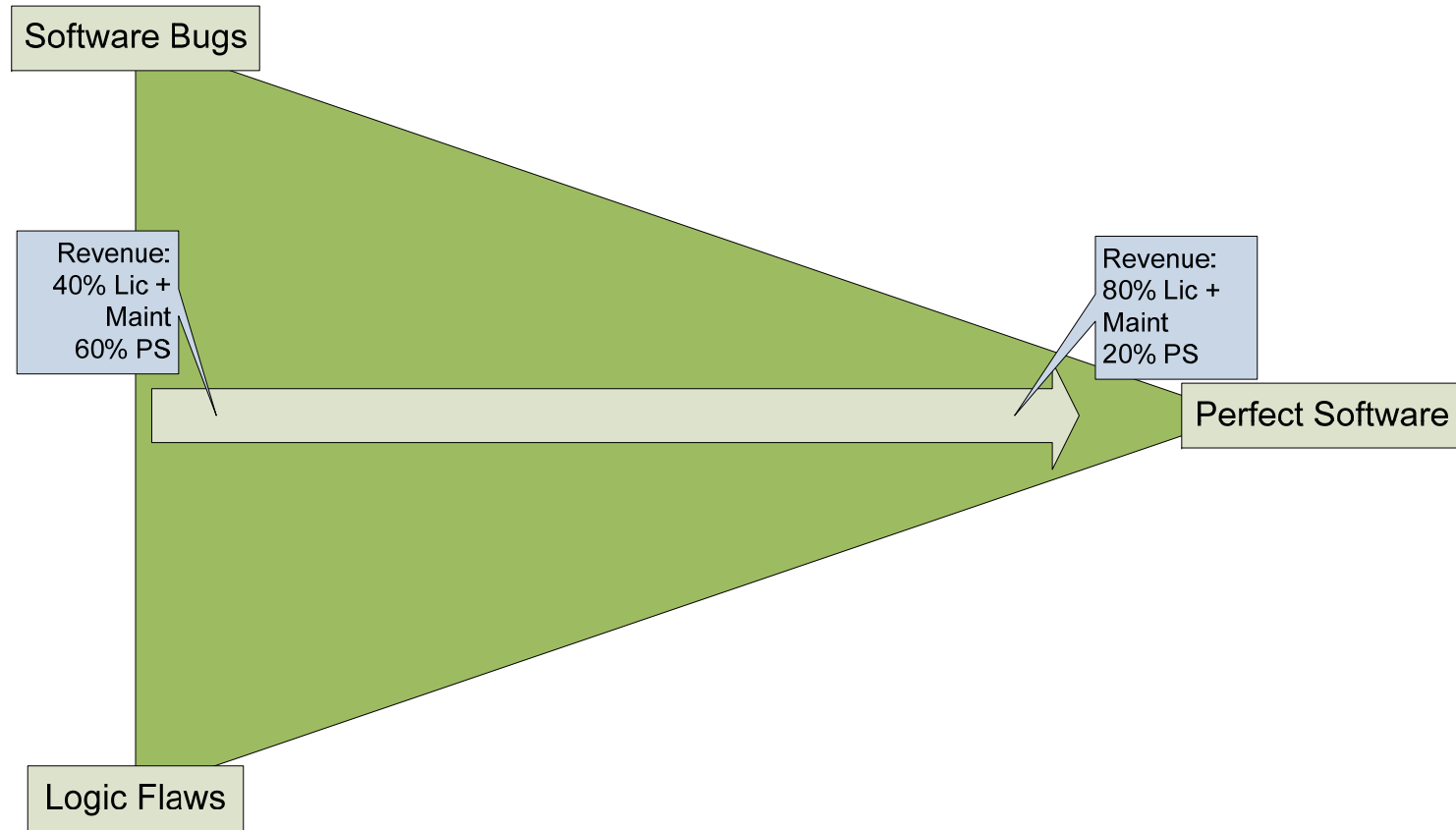
- ...but unfortunately there are 2 questions
  - Does the product work?
  - Are there resources to deploy?



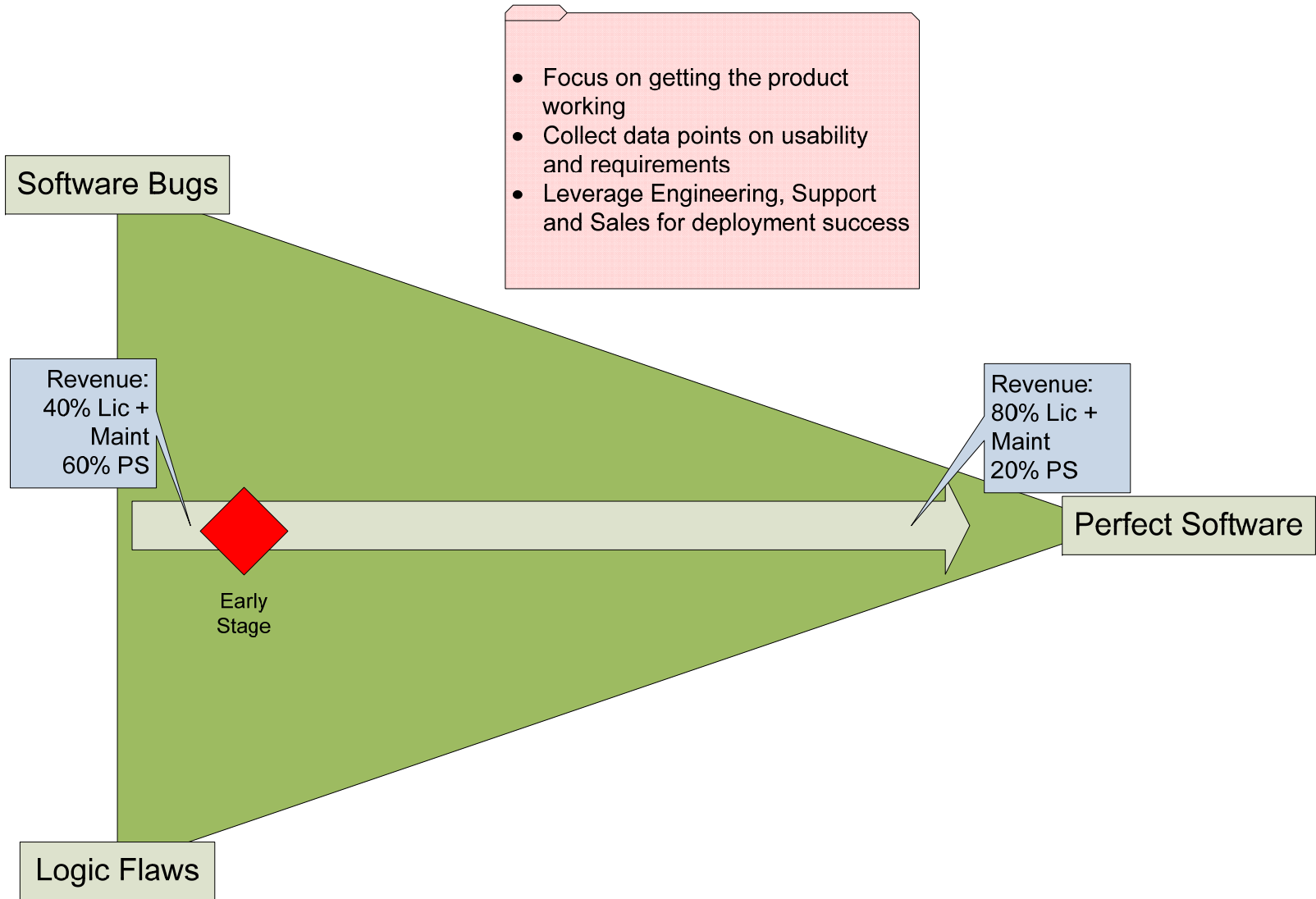
# Features of Early Stage Software Products

-  **The Good**
  - Serves a purpose – saves money / generates revenue
-  **The Bad**
  - Contains software bugs
-  **The Ugly**
  - Contains logic flaws

# The Balance between Bugs, Flaws and Perfect Software

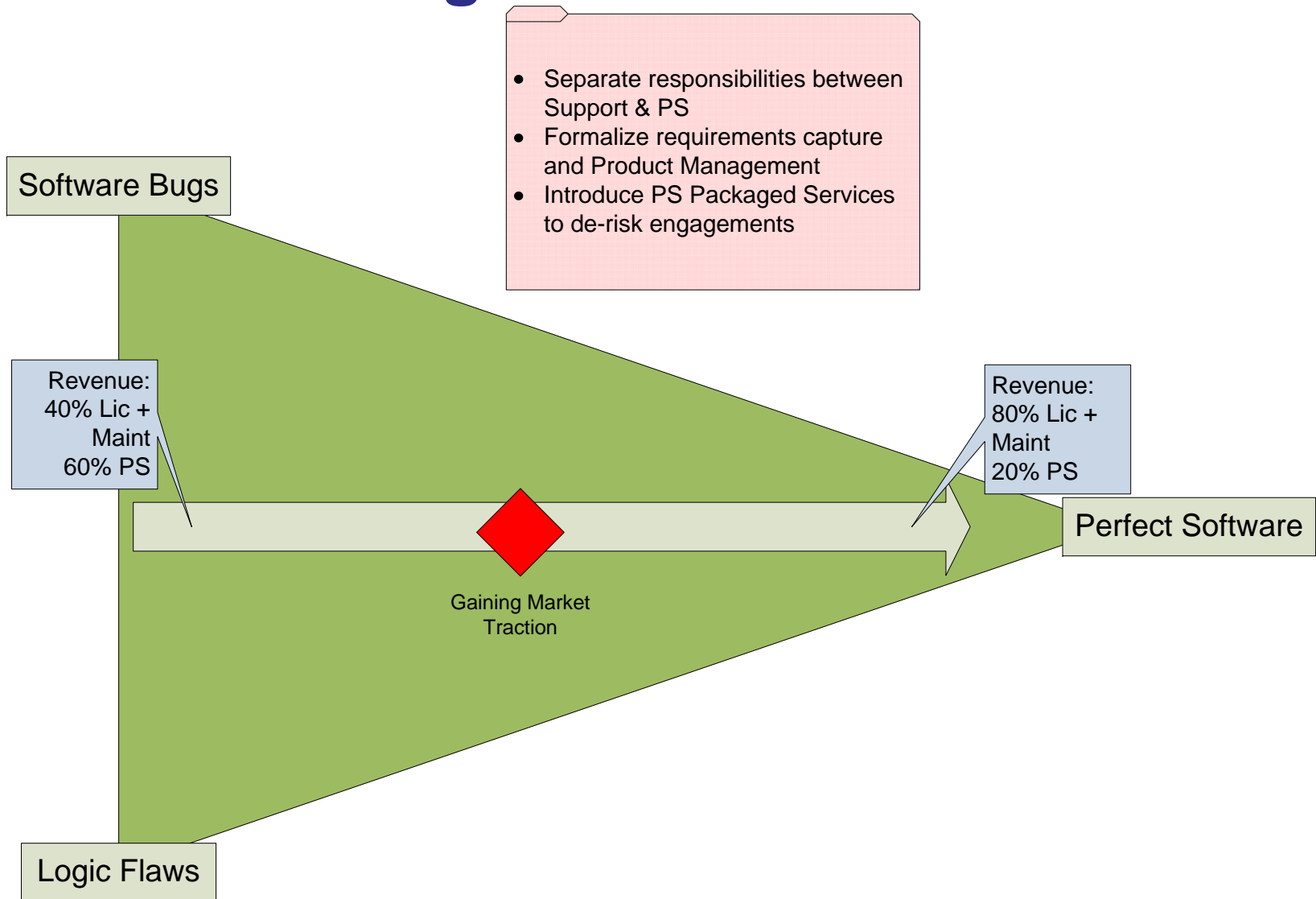


# The Position of Early Stage Companies

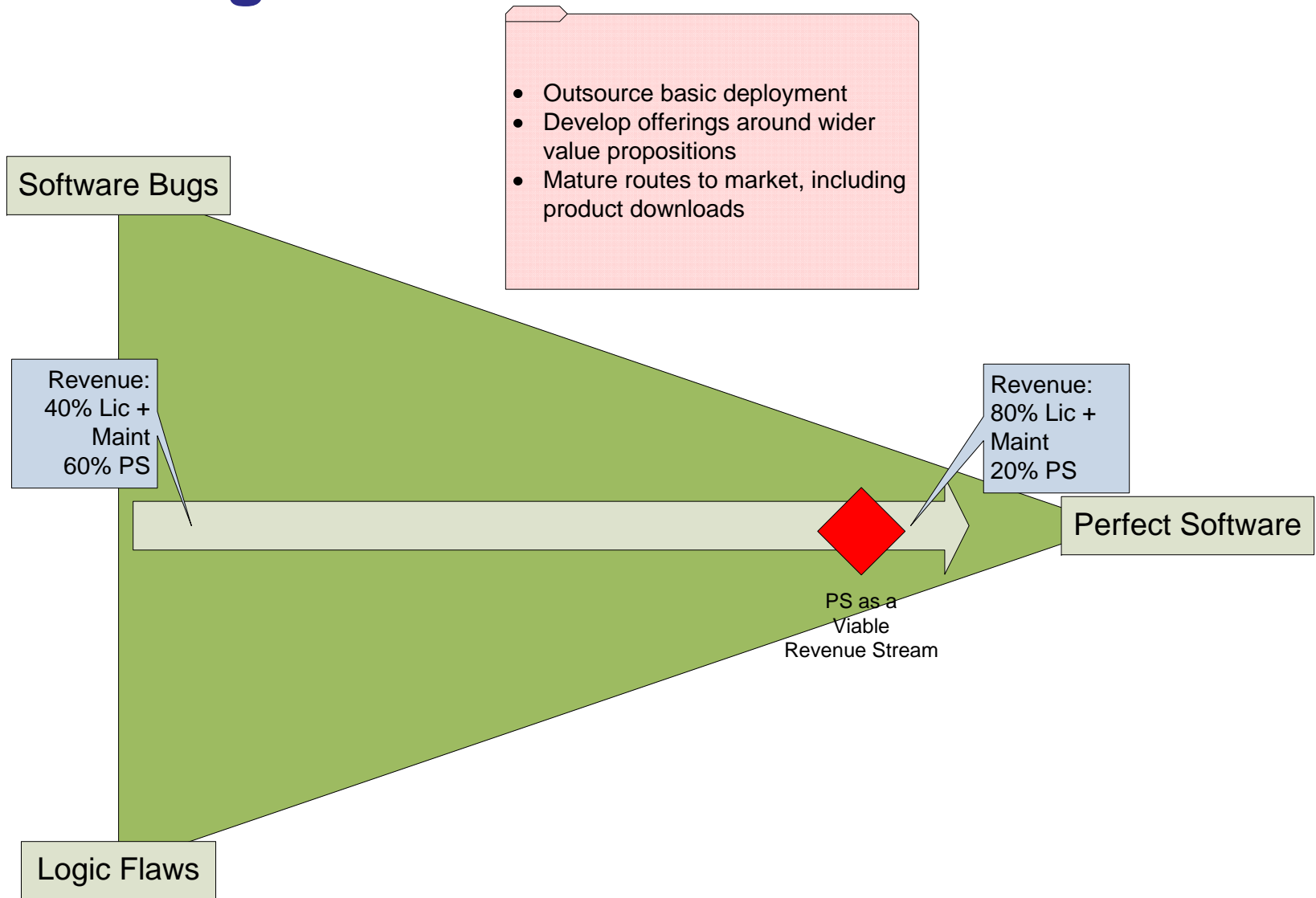




# ...Transitioning



# The Target



# On a Typical Engagement – 1/3?

- Early Stage:

- 80% of Effort Deploying the Product!

- 5% Product Setup & Modeling Applications
    - 35% Working around product bugs/flaws
    - 20% Meeting customer policy requirements
    - 40% Deploying Credentials

- 20% of Effort Advising the Customer!

- 30% Demonstrating where the discovered data adds value
    - 70% Learning product and deployment lessons

# On a Typical Engagement – 2/3?

- Transition:

- 50% of Effort Deploying the Product!

- 30% Product Setup & Integrations & Credentials
    - 15% Working around product bugs/flaws
    - 15% Meeting customer policy requirements
    - 40% Modeling Applications

- 50% of Effort Advising the Customer/Partner!

- 60% Demonstrating where the discovered data adds value
    - 40% Building revenue generating solutions

# On a Typical Engagement – 3/3?

- Target Model:

- 20% of Effort Deploying the Product!

- 20% Product Setup & Integrations & Credentials
    - 15% Working around product bugs/flaws
    - 45% Meeting customer policy requirements
    - 20% Modeling Applications

- 80% of Effort Advising the Customer/Partner!

- 100% Demonstrating where the discovered data adds value

# Great Product

- ...but unfortunately there are 2 questions
  - Does the product work?
  - **Are there resources to deploy?**

# Today's PS Challenges – 1/2

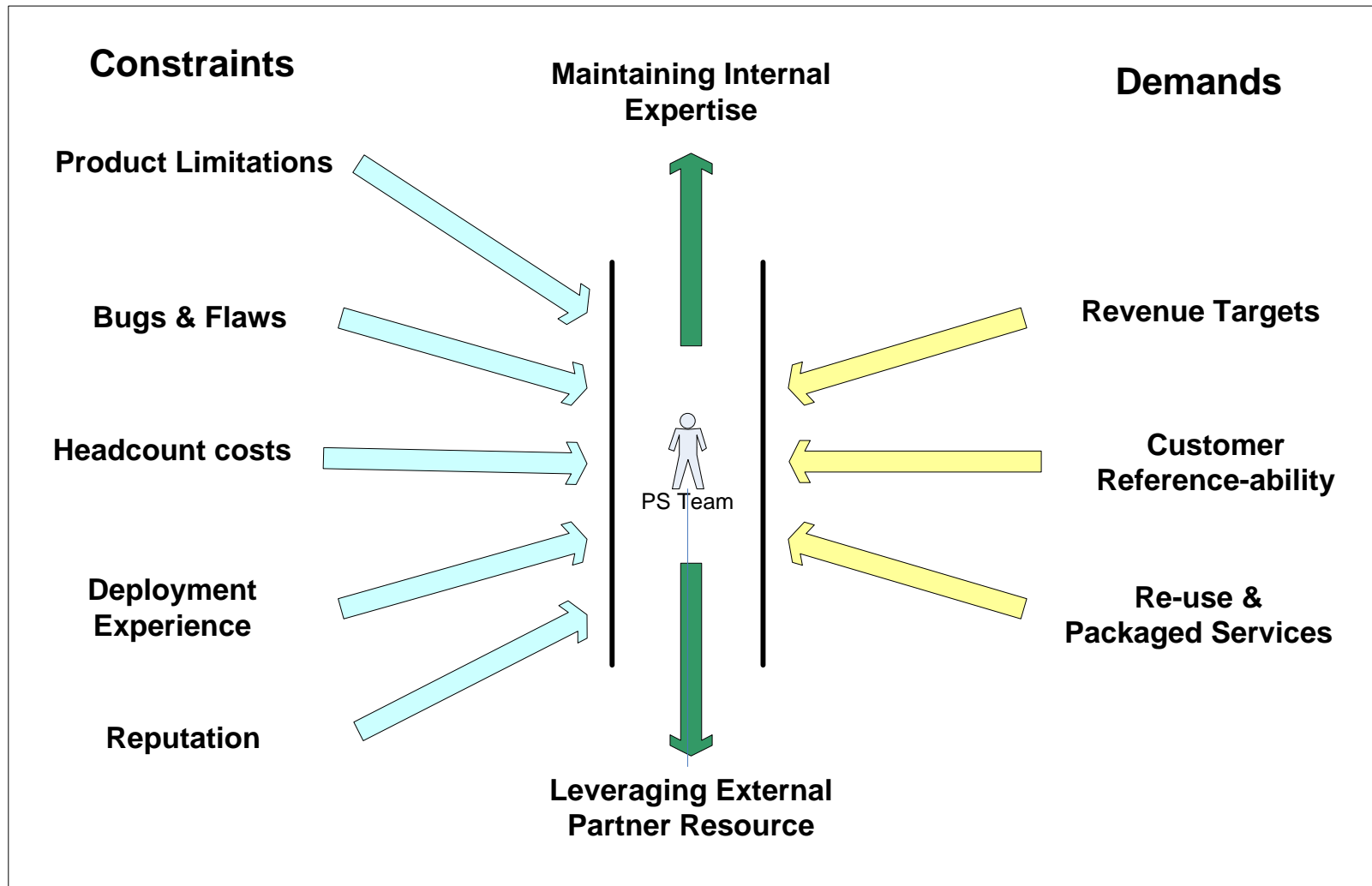
- Product Marketing
  - Product does 80% of what Marketing say
  - Product does 50% of what CEO says
  - Product is creating a new category
- Solution – Not shelf ware
  - Every customer's environment is special
  - Change impacts culture, people & process
  - No one extends a purchase on shelf ware
- Transferring Value
  - Business case for software purchase doesn't exist anymore
  - Customer wants to use – not deploy
  - Value is transferred through understanding

# Today's PS Challenges – 2/2

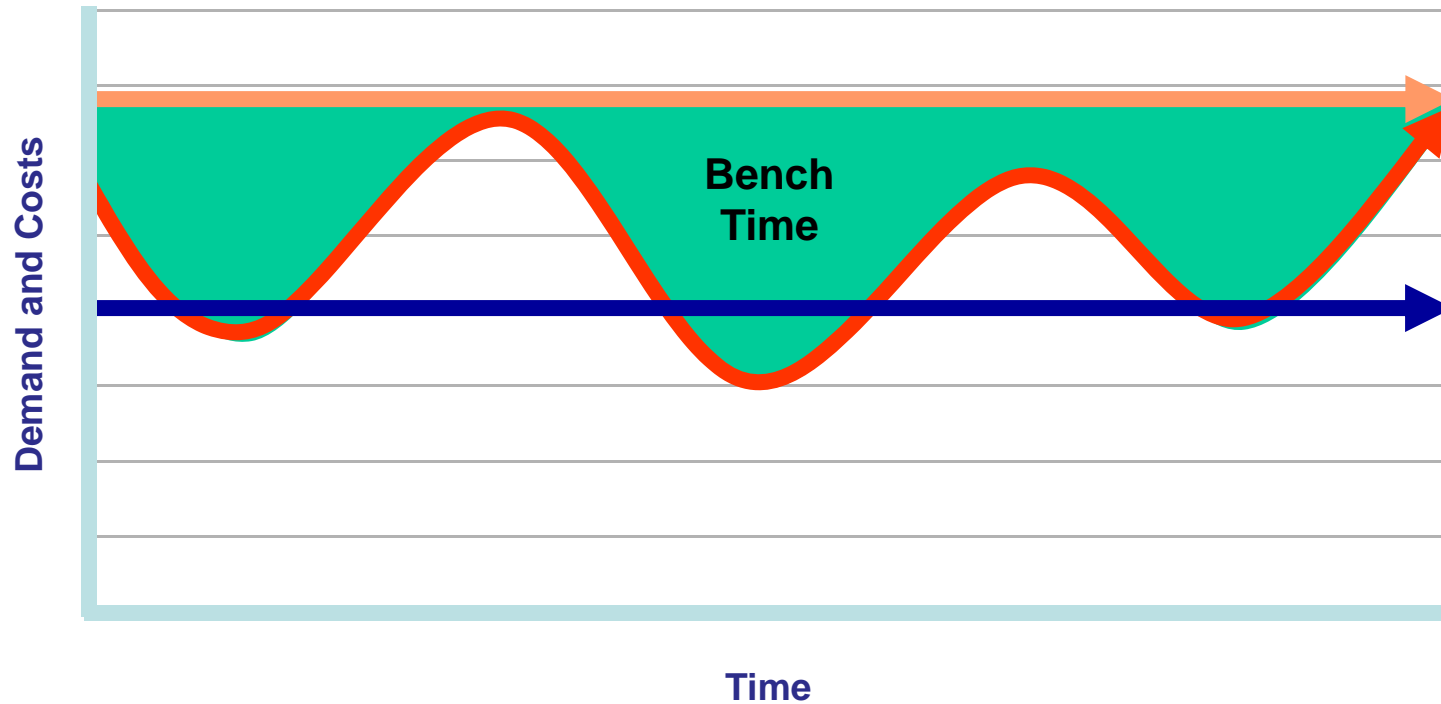
- Technical Challenges
  - Blockers which prevent progress
  - The product doesn't do that - yet
- The Competition
  - Staying one step ahead of the customer
  - Staying one step behind Product Engineering
  - Thou shall do what the external competition does – really?
- Profitability & Margin
  - PMs & TCs cost money
  - Revenue model – 80 % license + 20% Services
  - Sales will give Services away for free



# Constraints/Demands on The PS Team



# Economics of the “Allocation Problem”



**Customer Demand**

**Static Allocation Cost Model**

**Billable Waste**

**Dynamic Delivery Team Model**

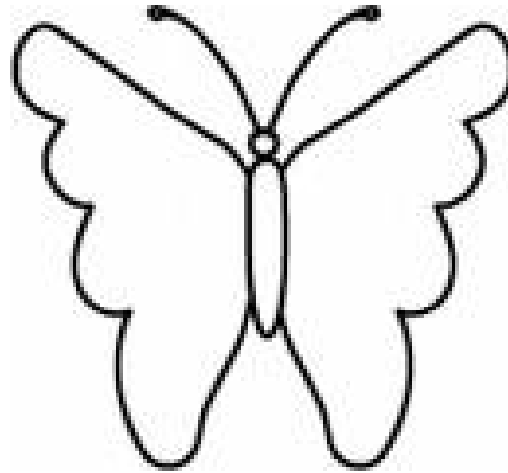
# The PS Butterfly Partnership

- Sales

- Product Management

- Engineering

- Support



- VARs

- Technology Partners

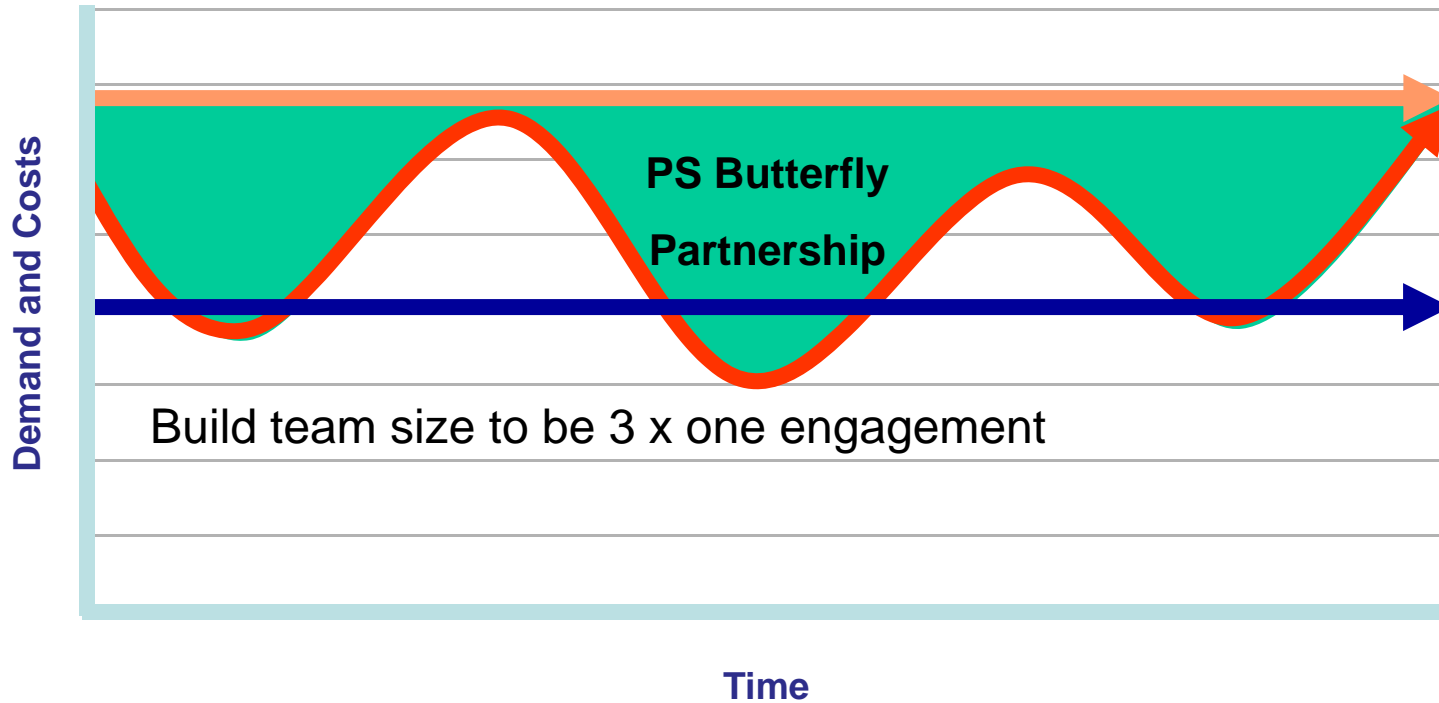
- Consultancy

- PS Body Shop

# Engagement Framework



# Team Size Spectrum



**Customer Demand**



**Leverage Internal Resource**



**Build Packaged Services**



**Partner Network**

# The Solution

- Focus on getting the product working
- Charge Premium Rates for PS time
- Maintain a fixed team size which transitions from deployment experts to value experts
- Grow PS by enabling Partners and Customers

# And Finally....Think Like an Entrepreneur

- Use other peoples' resources  
*Don't build out a large PS team based on quota carrier revenue targets, or forecasted backlog, but be ready based on the resources you have access to.*
- Enthusiasm goes a long way  
*PS Body shops get excited about revenue through billable hours only. Instead focus on those partners who are enthusiastic about what the product can achieve.*
- Let the CFO worry about cash-flow  
*Focus on reference-ability.*
- Know your product inside out  
*Be an extension of the selling machine – your on the same team.*
- Think Value!  
*When there is an exchange of value, the \$\$\$\$s follow. Focus on enabling your partners and customers – treat them the same – Enable through knowledge share and learning.*

# Is Your PS Team Ready?

## Questions for your PS Organization:

- Do you fully understand the current mix of bugs & flaws in your product or service and what this means to delivery success?
- Can you leverage internal virtual teams and external partner teams?
- Are you focused on enabling partners to sell and deliver for you?
- Can you monitor reference-ability as well as billability and utilization?
- Are you an 'enabling' organization?
- Are you ready for Self-Service product downloads





# Questions?

