



# Measurement-based Alignment of IT Strategies and Business Goals

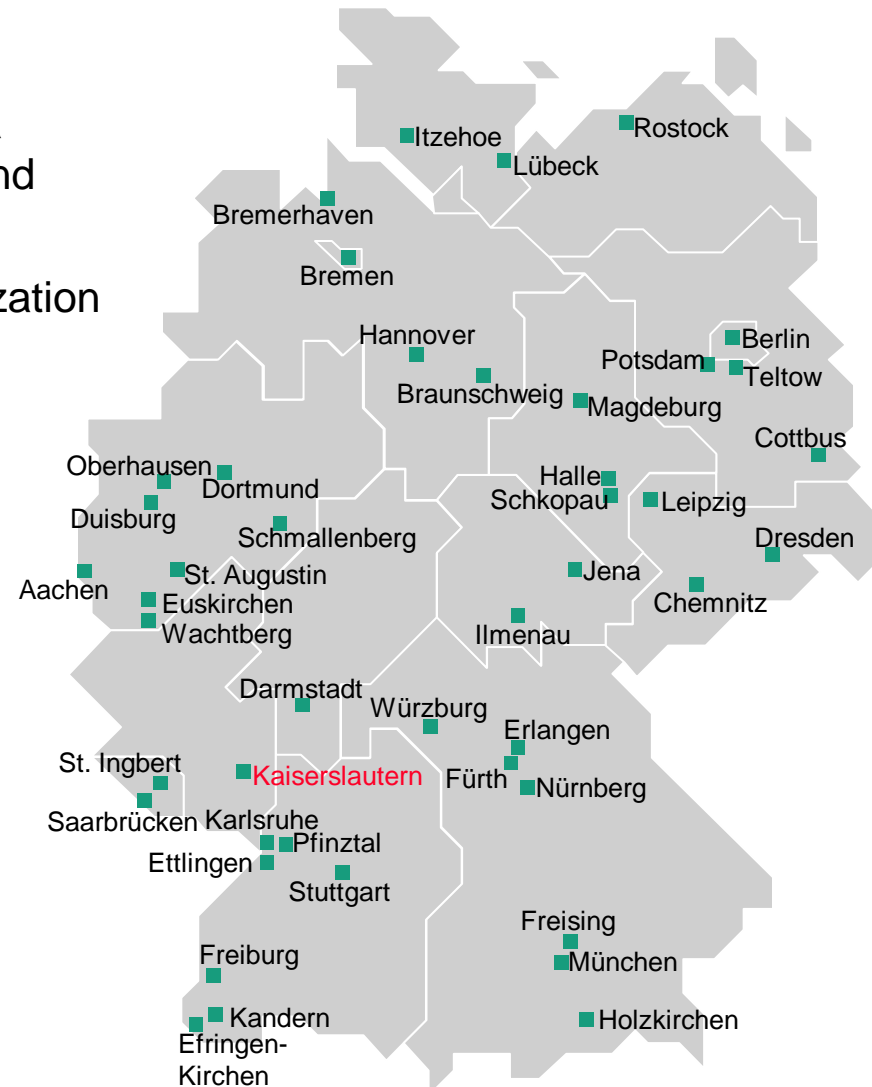
Presented by  
Dr. Jens Heidrich  
Dr. Adam Trendowicz

Authors of the GQM+Strategies® Method  
Dr. Victor R. Basili, Dr. Jens Heidrich, Dr. Mikael Lindvall,  
Dr. Jürgen Münch, Myrna Regardie, Dr. Dieter Rombach,  
Dr. Carolyn Seaman, Dr. Adam Trendowicz



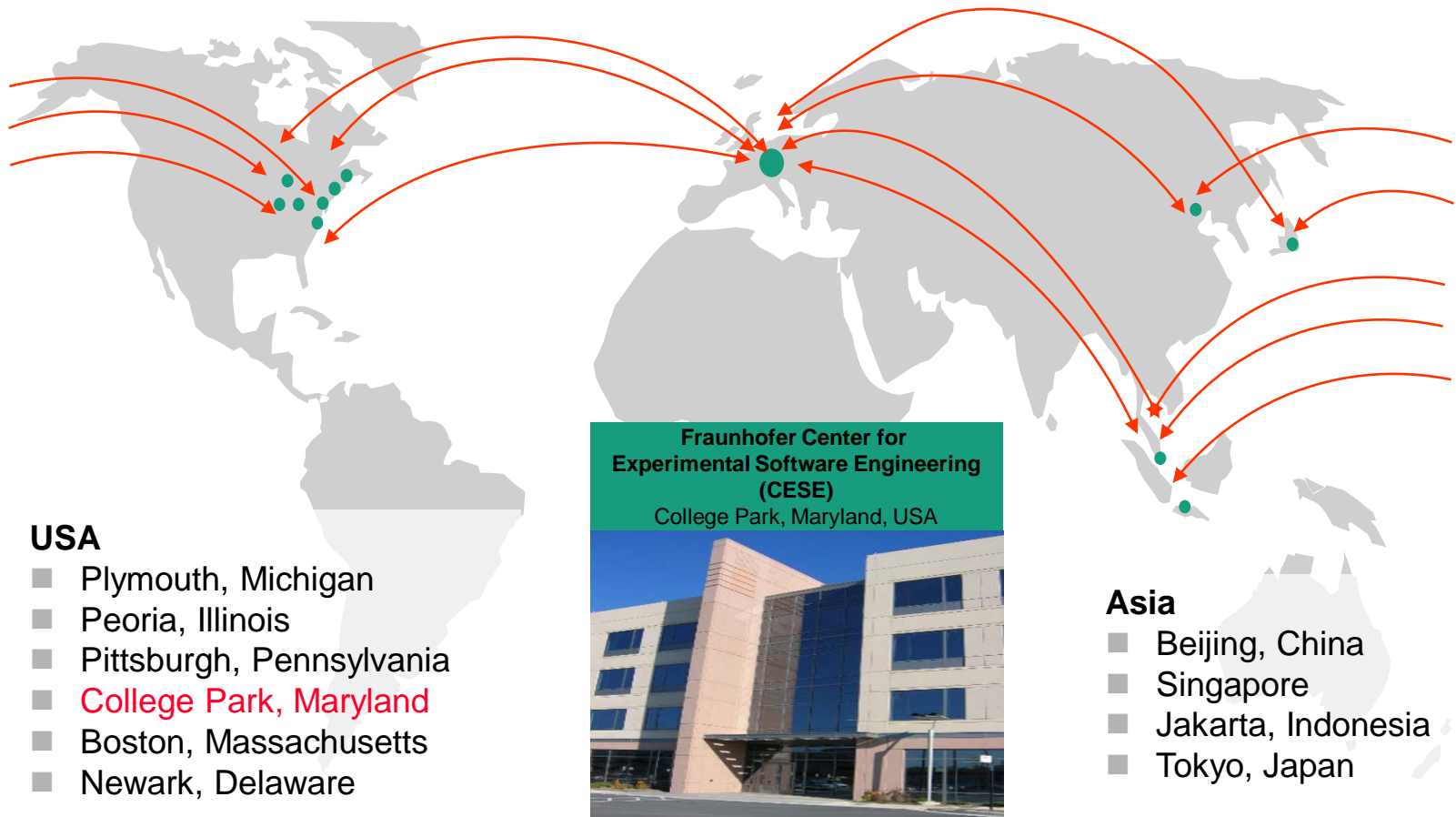
# About the Fraunhofer Gesellschaft

- Named after **Joseph von Fraunhofer** (1787-1826), a researcher and inventor and entrepreneur
- Germany's leading organization for **applied research and technology transfer**
- 59 institutes
- 17 000 employees





# Fraunhofer Locations Worldwide





# About the Presenters

## Dr. Jens Heidrich

- Head of  
Processes, Measurement, and  
Improvement (PMI)  
Department at  
Fraunhofer IESE
  
- Research focus
  - ☐ Goal-oriented Measurement
  - ☐ SW Project Control Centers
  - ☐ Quality Modeling
  - ☐ Software Estimation





# About the Presenters

## Dr. Adam Trendowicz

- Senior Engineer at  
Processes, Measurement, and  
Improvement (PMI)  
Department at  
Fraunhofer IESE
  
- Research focus
  - ☐ Software Estimation
  - ☐ Quality Modeling and  
Assessment
  - ☐ Goal-oriented Measurement
  - ☐ Quantitative Methods





## Brief Introduction Round of Tutorial Participants

- Name
- Position in your organization





# Agenda

---

## Session I

- 60 Minutes
- Introduction and Motivation
  - Goal-oriented Software Measurement
- 

## Session II

- 90 Minutes
- GQM+Strategies<sup>®</sup> Approach
  - Conclusions and Summary
-



# Measurement-based Alignment of IT Strategies and Business Goals

- **Introduction and Motivation**
- Goal-oriented Software Measurement
- GQM+Strategies® Method
- Conclusions and Summary



**Fraunhofer**

IESE



**Fraunhofer**

USA





# Importance of IT Business Alignment

- Software-intensive systems/services are taking over ever more tasks today
- IT and software becomes a central driver for innovation and growth
- Business success becomes dependent on IT/software-related strategies
- It is important to align IT/software-related strategies with business goals across the whole organization
  - Derivation of IT/software-related strategies from business goals
  - Illustrate contribution of IT/software-related strategies to business goals
- Otherwise, ...
  - IT/software is seen as a pure cost driver that is easy to substitute
  - Core competences for business success are outsourced



## Statements Underlining the Importance

- **IT investments needs to be aligned with business value**
  - “A real differentiator in a company’s financial performance is not overall IT spending but excellence in focusing this spending by business value.”  
(Accenture, 2004)
- **Alignment is hard to achieve in practice**
  - In Forrester’s IT Excellence survey of 162 senior IT executives, only 15% declared themselves to be fully aligned with the business.  
(Forrester, 2007)
- **Alignment is strategic management priority**
  - A 2006 survey of over 1,400 CIOs from around the world noted that linking business strategy with IT planning is the second most important strategic management priority for 2006 through 2009.  
(Gartner, 2007)

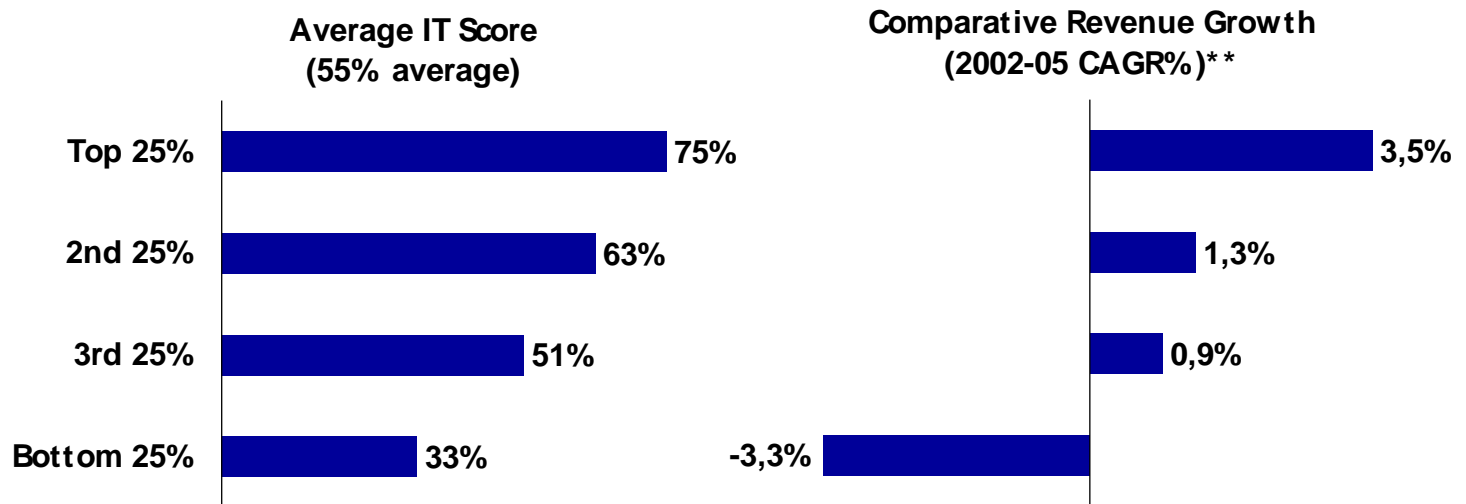


## ROI of IT Investments

- **Proper IT investments** (in terms of visibility, control, and productivity) make a quantifiable, positive difference in business performance

### IT Score and Three-Year Revenue Growth\*

161 manufacturing enterprises in U.S., Japan, and Western Europe



\*Statistically significant to the 99% level

\*\*Compared to peer enterprises in the same industry sector.

(Source: Keystone, 2006)



## Critical Questions

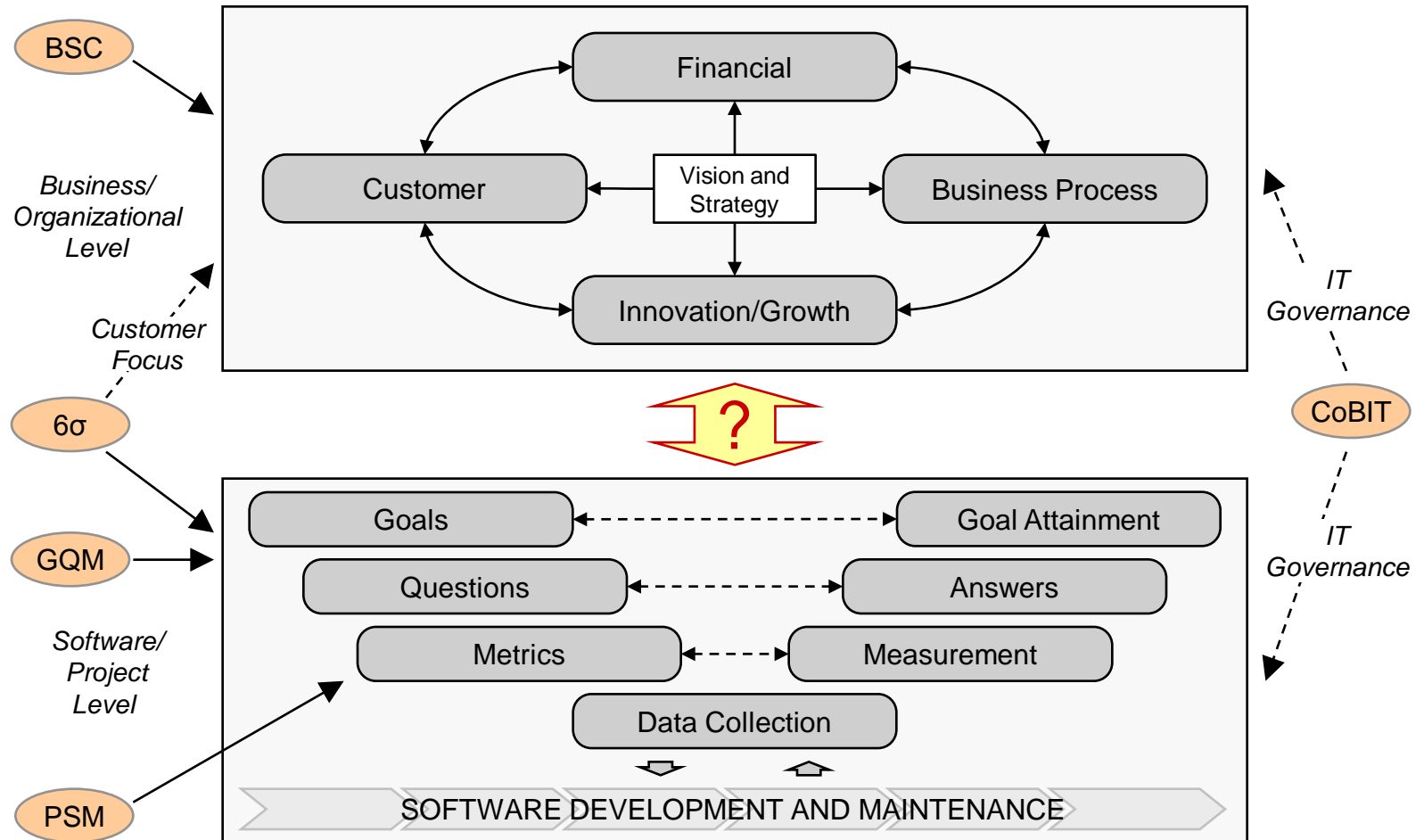
- How can I use IT to reduce costs and sustain growth?
- How much money should I spend on IT?
- How do IT investments contribute to my business value?
- What are critical factors that affect my business goals?
- How to avoid unrealistic and contradictory goals/strategies?
- What data do I need for guiding business improvement?

In practice, these questions are hard to answer because

- there is **no explicit linkage** between business goals and IT-related strategies
- there are **limited measures** in place for evaluating whether and how applied IT solutions generate business value



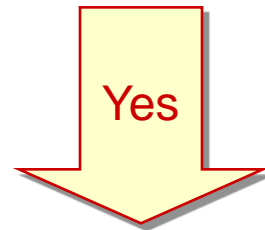
# Management Gap





## Closing the Management Gap

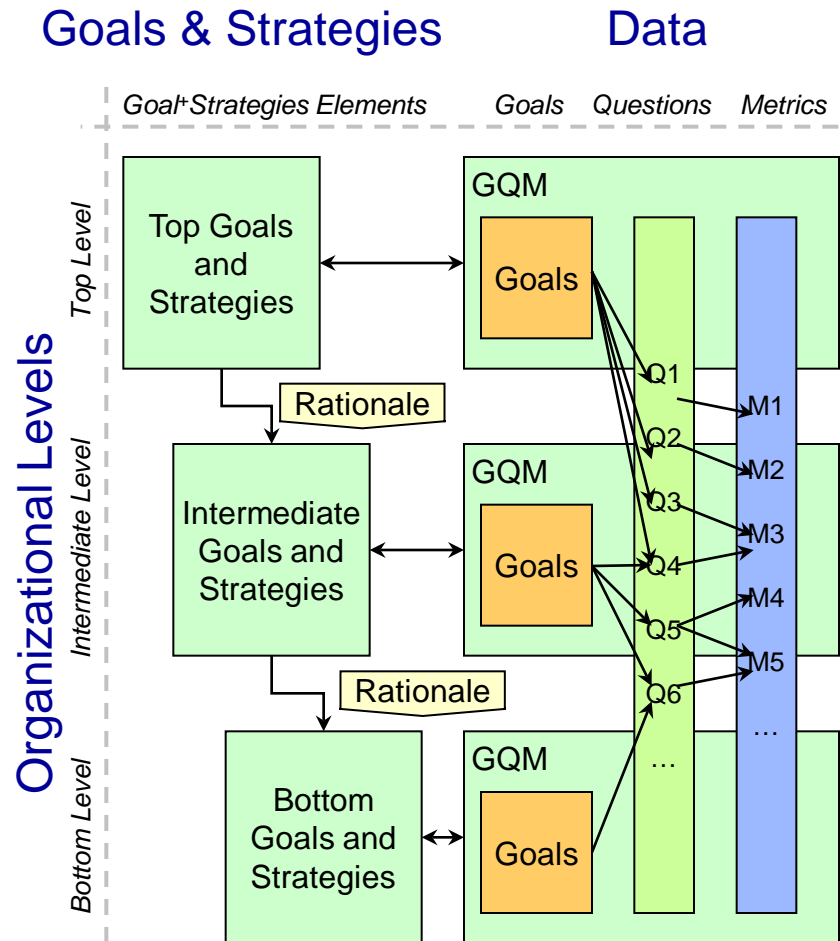
- Can we translate business vision and strategy into a set of operational goals and corresponding quantitative management?
- Can we explain how meeting certain objectives on the lower levels contributes to business-level objectives?





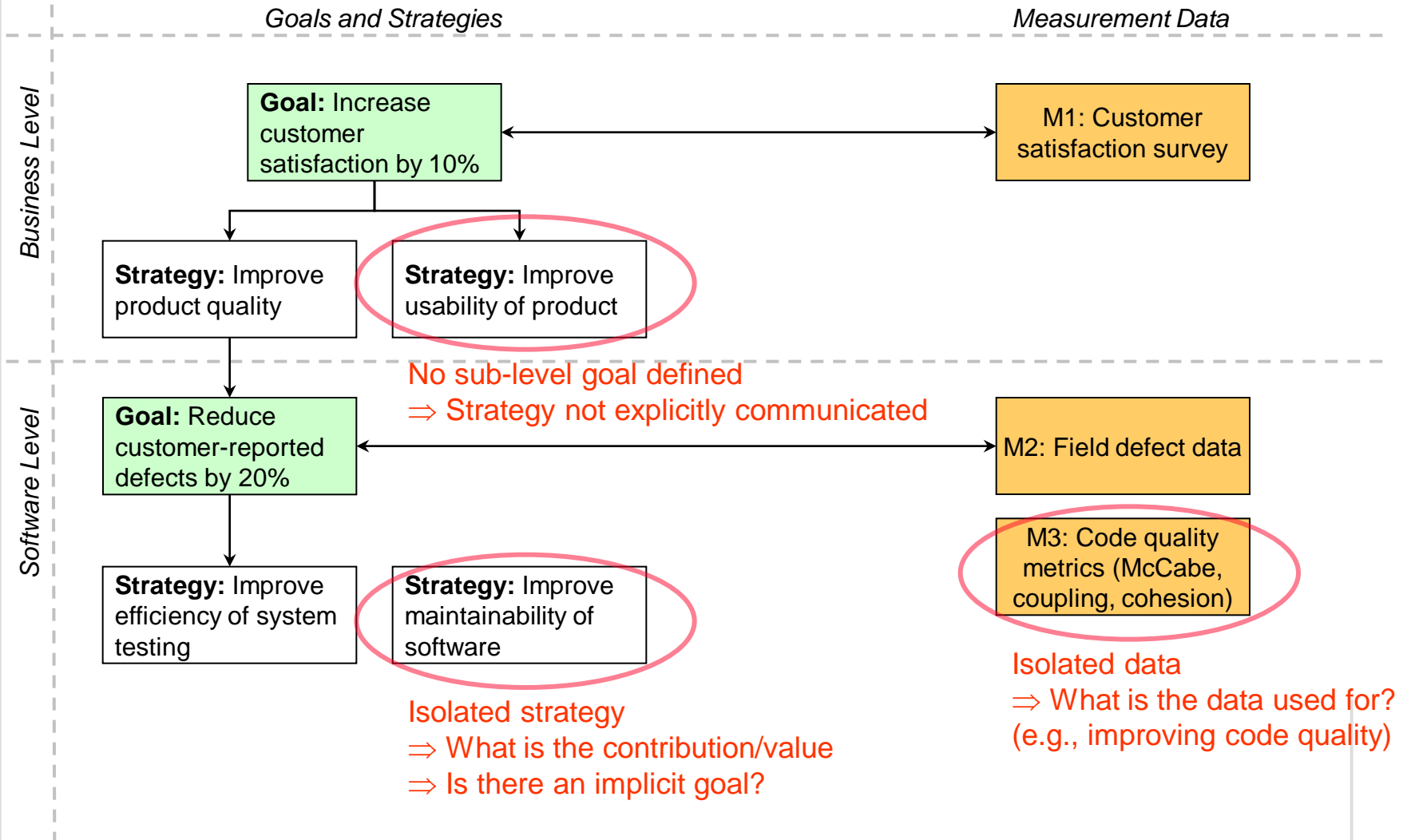
# Closing the Management Gap

- Align the business at all levels of the organization
- Link goals and strategies from the top management level down to the project level
- Control success/failure of goals and strategies through measurement
- Document the rationale for linking goals and strategies (context and assumptions)
- Close gaps and let all goals and measurement data contribute to a consistent and meaningful story





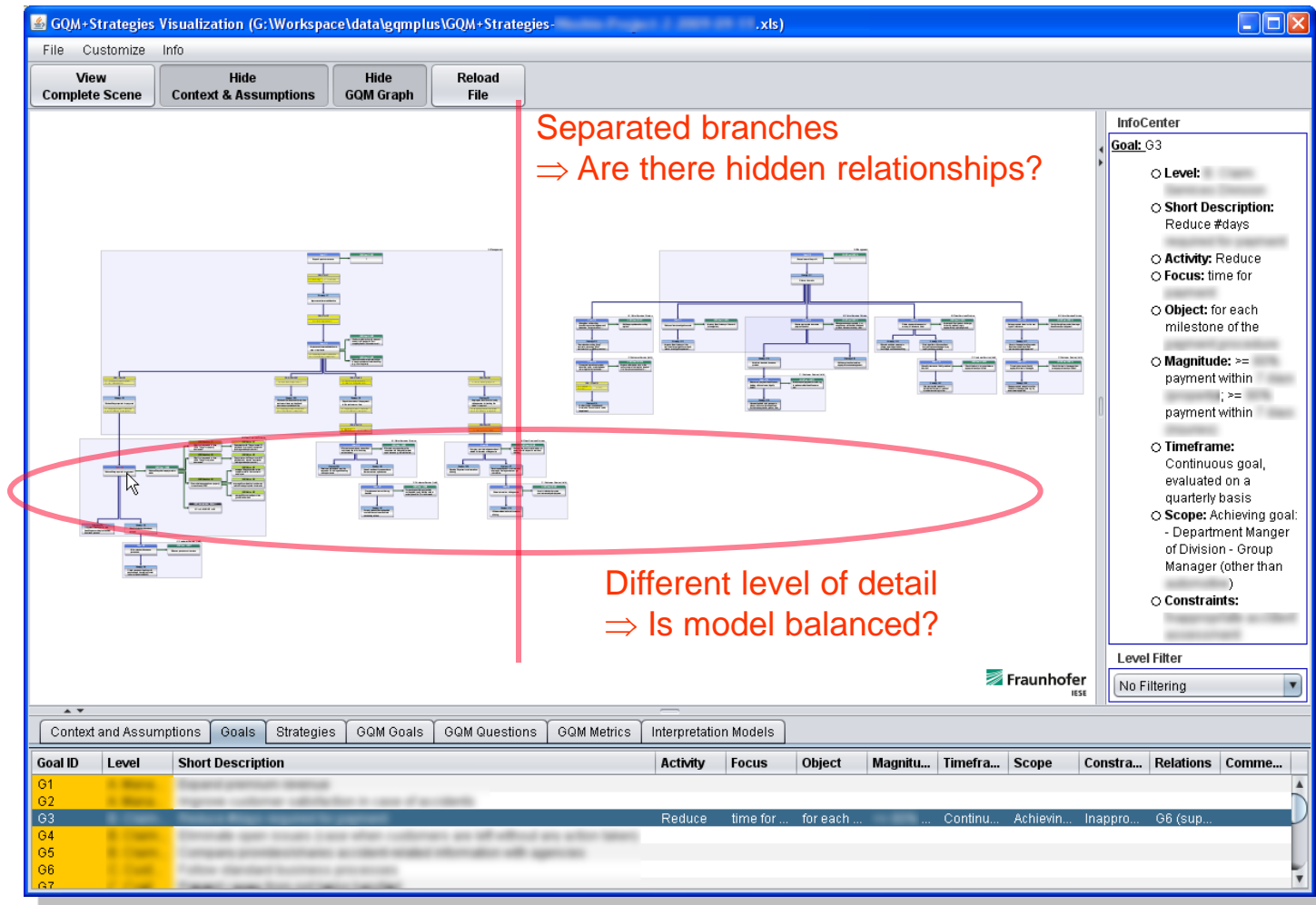
## Gap Analysis Example (1/2)







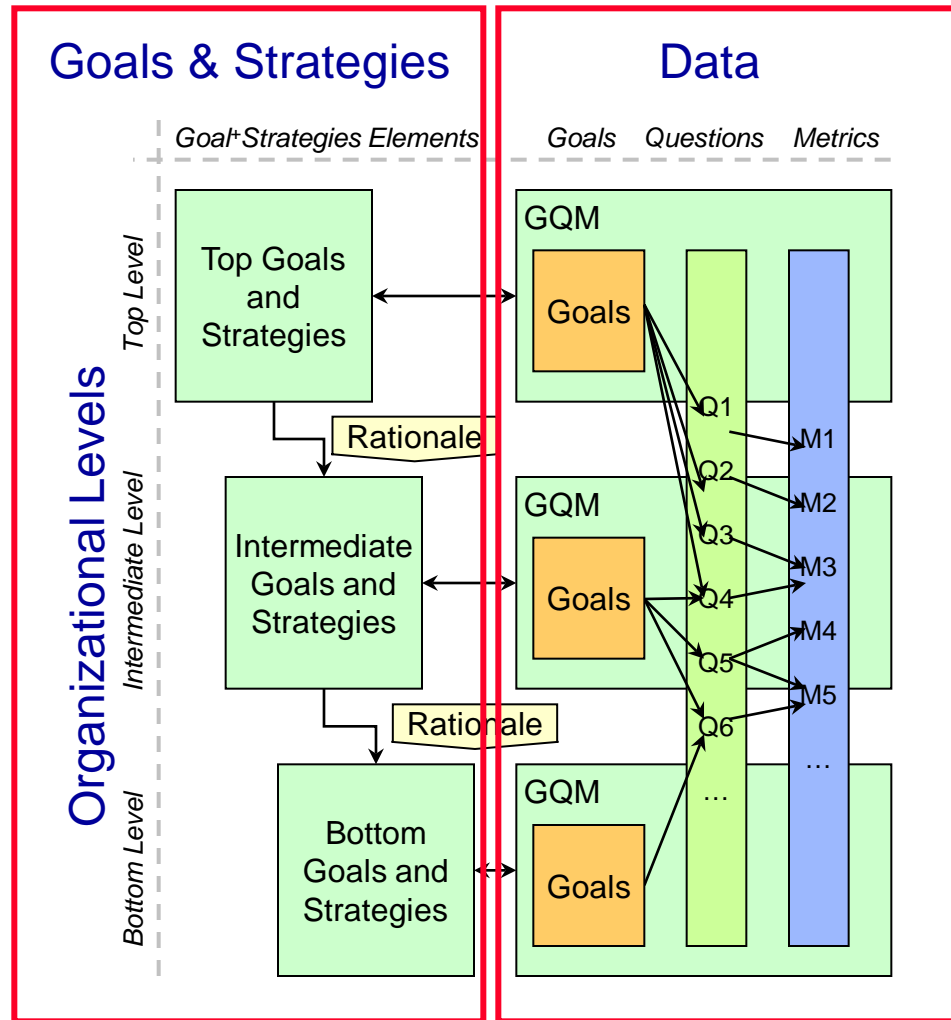
## Gap Analysis Example (2/2)





# Closing the Management Gap

Chapter 3:  
GQM+Strategies®  
Method



Chapter 2:  
Goal-oriented  
Software  
Measurement



## Measurement-based Alignment of IT Strategies and Business Goals

- Introduction and Motivation
- **Goal-oriented Software Measurement**
- GQM+Strategies® Method
- Conclusions and Summary






# Why Do Organizations Measure?

- **Understand the Business and Create Visibility**
  - ☐ Build baselines, show relationships
  - ☐ Identify critical factors
- **Manage and Control Projects Based on Quantitative Evidence**
  - ☐ Plan and estimate
  - ☐ Track- actuals versus estimates
  - ☐ Decision-making
- **Guide Improvement and Optimize the Activities**
  - ☐ Prioritize
  - ☐ Assess
  - ☐ Package Experiences

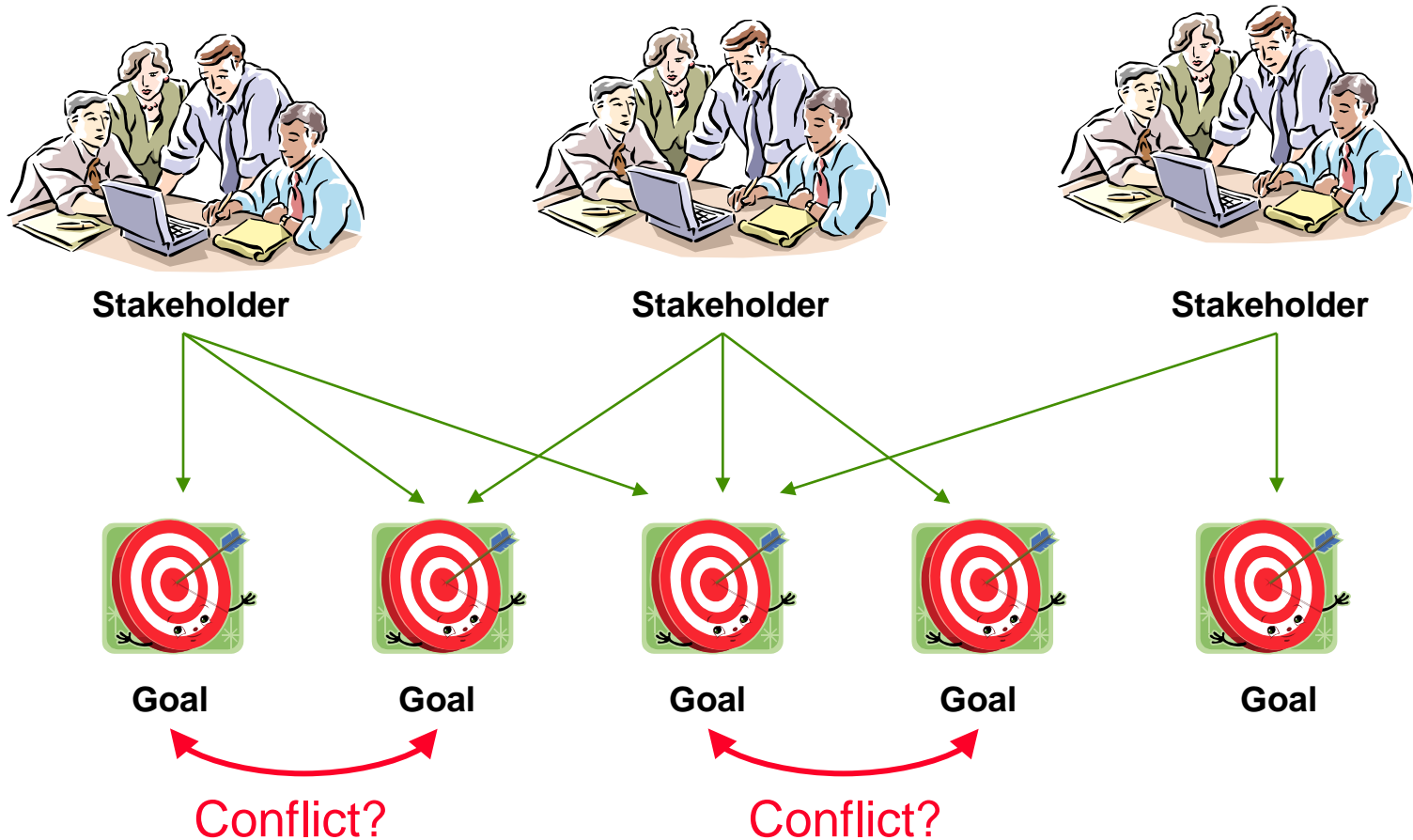
**Measurement is a means to an end, not an end in itself**



# What is Measurement? Quantifying Entities

Entities	Attributes	Rules	Numbers/Symbols
 Process	effort	person-days	53 pds
 Product	size	number of Lines of Code	700 LOC
 Resource	experience	>10 projects	“high”

# Internal and External Stakeholders have their Own Goals





## Levels of Measurement Ability

Ability Level	Level	Description	Ability
	5: Motivate / Improve	Describe what needs to be done to control and manage	<i>Build prescriptive models</i>
	4: Predict	Estimate expected product quality and process resource consumption	<i>Build predictive models</i>
	3: Evaluate	Assess achievement of quality goals, impact of technology on products	<i>Compare models</i>
	2: Understand	Explain associations / dependencies between processes and products Discover causal relationships	<i>Analyze models</i>
	1: Characterize	Describe and differentiate software processes and products	<i>Build descriptive models and baselines</i>



## Measurement is not just the collection of data...

calendar time

*total lines of code*

number of failures during system test

**number of open problems**    total effort

cyclomatic complexity

number of defects found in inspections

*severity of failures*

*lines of code/staff month*

total number of defects





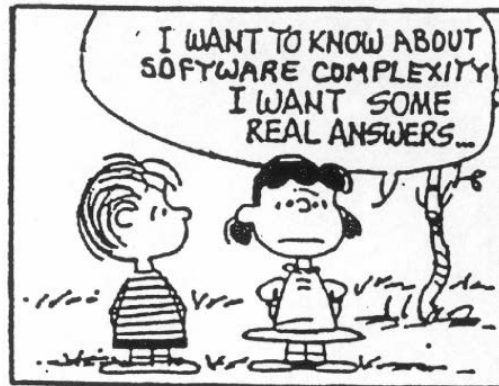
**... and cannot simply be aggregated...**



Copyright © 2007, William E. Riddle.



## ...and requires interpretation



Copyright © United Feature Syndicate, Inc.



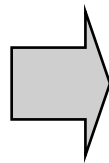
# Problems with Measurement

## ■ Problems

- ☐ Too much unnecessary data is collected
- ☐ Data is not analyzed
- ☐ Data is not analyzed in the right environment
- ☐ Standard measures are postulated without adaptation for the environment
- ☐ Important aspects cannot be analyzed because data is missing

## ■ General Consequences

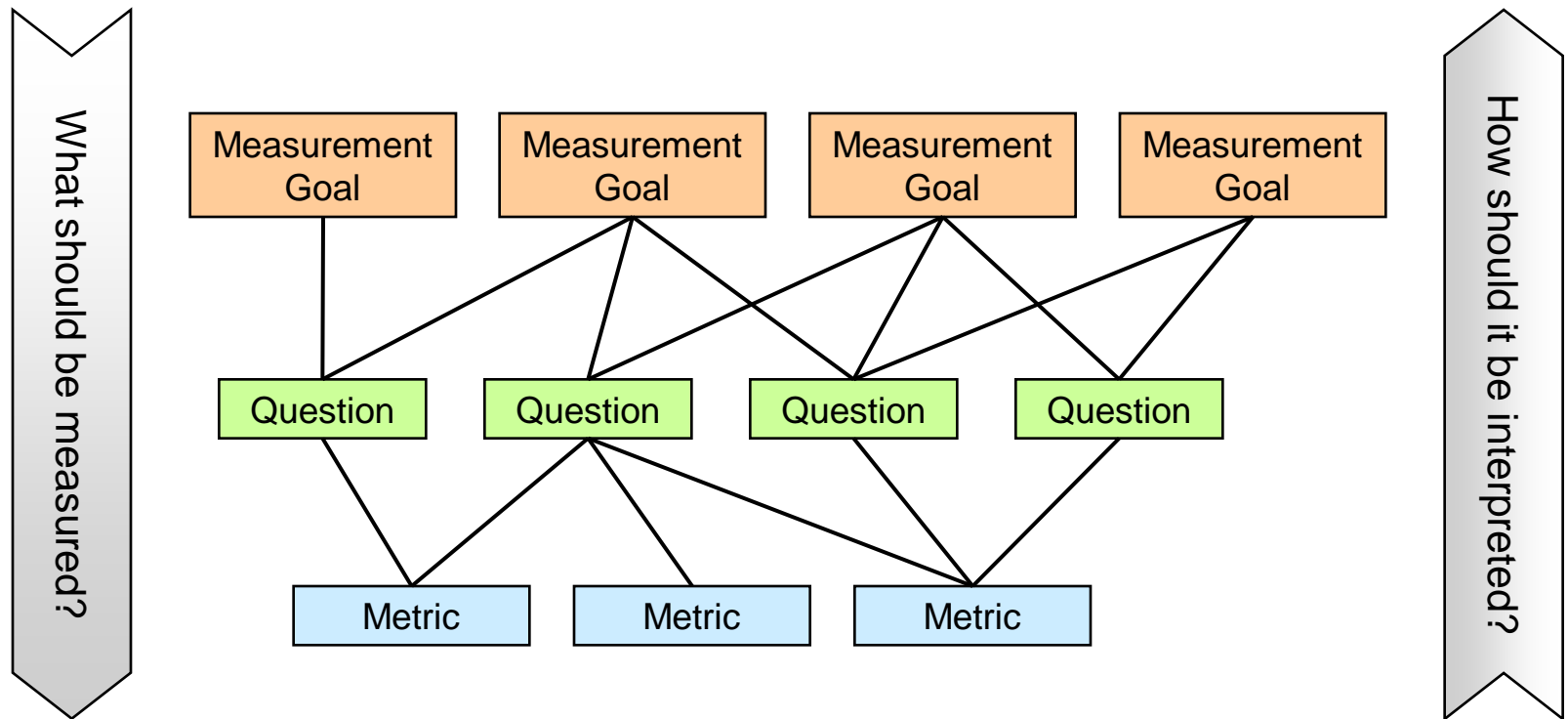
- ☐ Wrong conclusions can be drawn
- ☐ Insufficient pay-off for the cost



**Goal-oriented Measurement**



# The GQM Structure

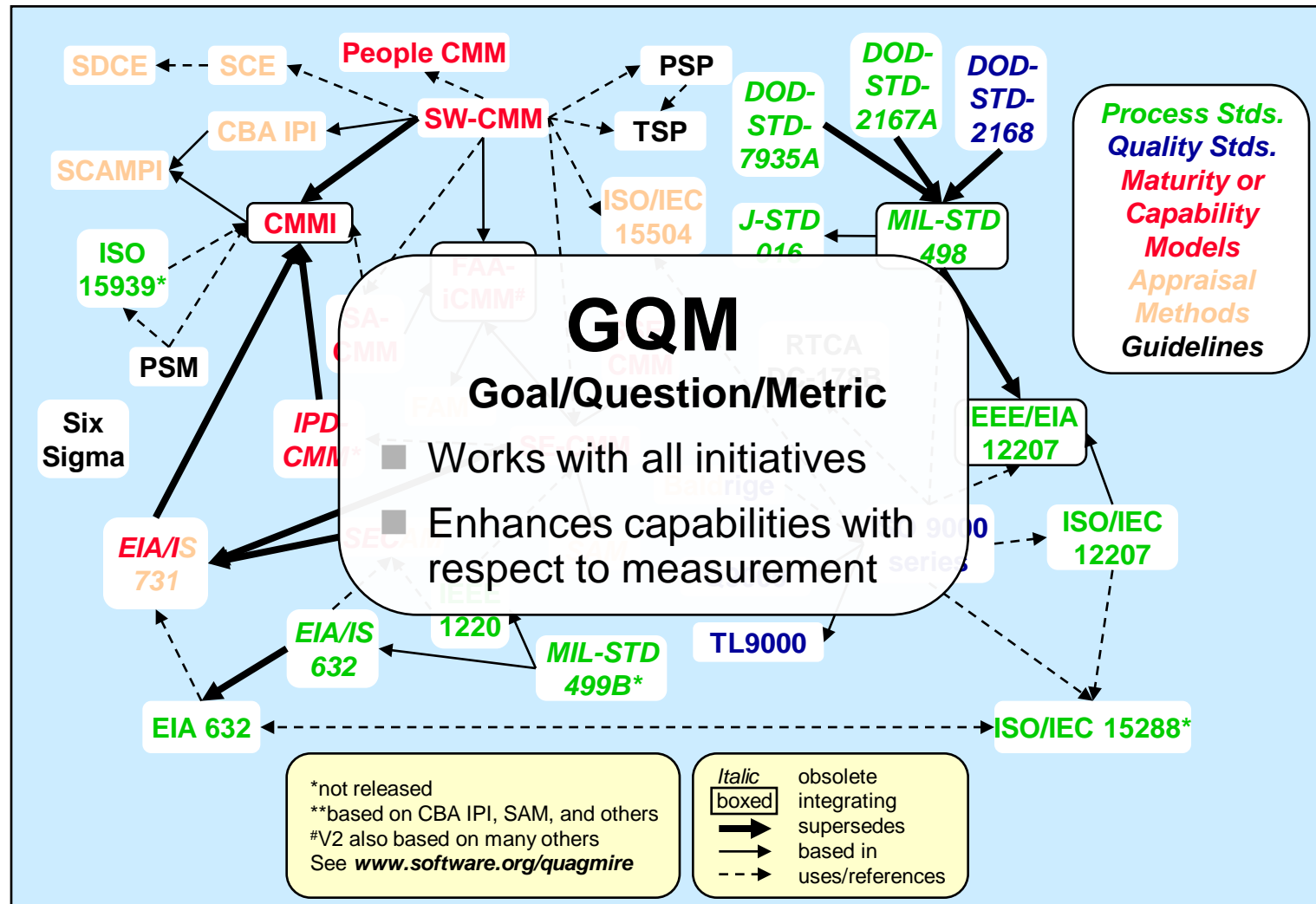




## Basics of the GQM Approach

- **Develop** a set of measurement **goals** for aspects of the software product and process of interest (e.g., productivity and quality)
- **Generate questions** (based upon models) that define those goals as completely as possible in a quantifiable way
- **Specify the measures** needed to be collected to answer those questions and track process and product conformance to the goals
- **Develop mechanisms for data collection**
- **Collect, validate, and analyze the data** in real time to provide feedback to projects for corrective action
- **Package knowledge** based on the interpretations and make recommendations for future improvements

## Measurement and other Quality Initiatives





# Goal Generation Template

- Goals may be defined for any object, for a variety of reasons, with respect to various models of quality, from various points of view, relative to a particular environment.
  - **Analyze some**
    - Object of study: process, product, resource, ...
  - **for the purpose of**
    - Purpose: characterize, evaluate, predict, motivate, improve
  - **with respect to**
    - Focus: cost, correctness, defect removal, changes, reliability, user friendliness, ...
  - **from the point of view of**
    - Stakeholder: user, customer, manager, developer, corporation, ...
  - **in the following context**
    - Context: problem factors, people factors, resource factors, process factors, ...



## Example 1: Generating a Measurement Goal

Consider the following situation:

- The organization needs to improve the quality of their products (**Business Goal**) because customers report too many failures, most of which should have been caught during the system test
- It is considering adopting a new system test process (a risk and expense) and wants to try the new system test process on a pilot project (**Strategy**) to determine if it is doable and more effective than what it has been doing (**Software Goal**)
- The organization has data on the number of faults identified by the system test process and the number released to the field for various products. It uses a waterfall type life cycle process, ... (**Context**)
- To make an informed decision it must define the new test process, determine if it is being followed, characterize how well the process is identifying faults, and compare it to what they were doing before (**Measurement Goal**)





## Example GQM Goal 1

<b>Object</b>	Analyze the system test process
<b>Purpose</b>	for the purpose of evaluation
<b>Focus</b>	with respect to defect slippage
<b>Viewpoint</b>	from the point of view of the corporation
<b>Context</b>	in the context of the specific organizational environment



# Guidelines for Deriving Process-Related Questions

## ■ Process Conformance

- ☐ Characterize the process quantitatively and assess how well the process is performed
- ☐ How do we quantitatively characterize the process?

## ■ Domain Understanding

- ☐ Characterize the object of the process and evaluate the knowledge of the object and its domain by the process performers
- ☐ How do we quantify this knowledge?

## ■ Focus

- ☐ What is the aspect of the process which is of interest?
- ☐ Analyze the output of the process according to some quality model and some viewpoint

## ■ Feedback

- ☐ What has been learned about the process, its application, the product domain, or any other process or product?



## Example 1: Generating Questions

### ■ Process Conformance

- ☐ Q1: What is the experience of the team with respect to the method?
- ☐ Q2: How many requirements are there?
- ☐ Q3: What is the importance of testing each requirement?
- ☐ Q4: What is the complexity of testing each requirement?
- ☐ Q5: What is the distribution of tests over requirements?
- ☐ Q6: Is the number of tests per requirement consistent with its complexity and importance?

### ■ Domain Understanding

- ☐ Q7: How familiar is the domain?
- ☐ Q8: How understandable are the requirements for this project?

### ■ Focus

- ☐ Q9: What is the defect slippage model and does the new process reduce it sufficiently to invest in it?

### ■ Feedback

- ☐ Q10: Can the process be improved?, ...



## Example 1: Measuring the Questions (1/2)

- **Q1:** What is the experience of the team with respect to the method?
  - **M1:** Subjective rating per person
- **Q2:** How many requirements are there?
  - **M2:** Overall number of requirements
- **Q3:** What is the importance of testing each requirement?
  - **M3:** Subjective rating by marketing and testers
- **Q4:** What is the complexity of testing each requirement?
  - **M4:** Subjective rating by testers
- **Q5:** What is the distribution of tests over requirements?
  - **M5:** Average number of tests per requirement
  - **M6:** Standard deviation from the average number
- **Q6:** Is the number of tests per requirement consistent with its complexity and importance?
  - **M7:** Subjective rating per person



## Example 1: Measuring the Questions (2/2)

- **Q7:** How familiar is the domain?
  - **M8:** Subjective rating per person
- **Q8:** How understandable are the requirements for this project?
  - **M9:** Subjective rating per person
- **Q9:** What is the defect slippage model and does the new process reduce it sufficiently to invest in it?
  - **M10:** Ratio of faults found in system test to the faults found after system test on this project (if at least 1 fault is found)
  - **M11:** Ratio of faults found in system test to the faults found after system test in the set of projects used as a basis for comparison (if at least 1 fault was found)
  - **DSR = M10/M11:** The Defect Slippage Rate (DSR) is defined as the relationship of system test on this project to faults as compared to the average of the appropriate basis set
- **Q10:** Can the process be improved?
  - ...



## Example 1: Interpreting the Defect Slippage Model

- If  $DSR > 1$ , then
  - ☐ Method better than history for this class of project
  - ☐ Check process conformance
  - ☐ If process conformance poor, improve process or process conformance
  - ☐ Check domain understanding
  - ☐ If domain understanding poor, improve object or domain training
- If  $DSR \sim 1$ , then
  - ☐ Method equivalent to history for this class of project
  - ☐ If cost lower than normal, method cost effective
  - ☐ Check process conformance ...
- If  $DSR < 1$ , then
  - ☐ Check process conformance
  - ☐ If process conformance good, check domain conformance
    - ☐ If domain understanding good
      - ☐ Method poorer than history for this class of project



## Advice and Considerations

- Measurement should not be an end in itself, but a **key factor to reach business goals**
- Measurement needs to be **deeply integrated** into organizational processes
- Measurement programs help to make **decision making more transparent**
- **Goal-oriented measurement** is the basis for the success of measurement programs
- **Higher-level goals** require more understanding, but have a bigger payback
- There is **no universal measurement program** solving all problems related to measurement



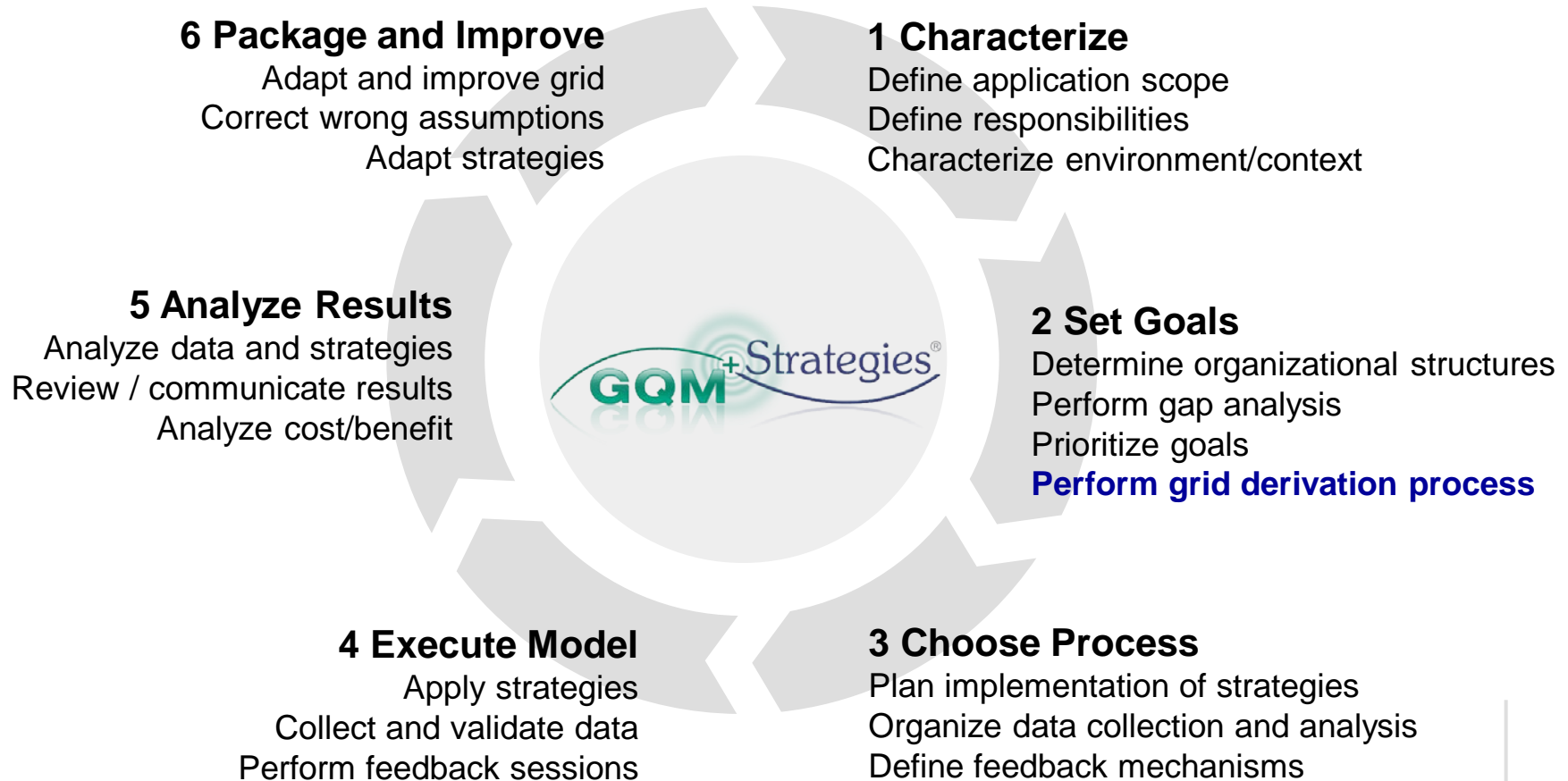
## Measurement-based Alignment of IT Strategies and Business Goals

- Introduction and Motivation
- Goal-oriented Software Measurement
- **GQM+Strategies® Method**
- Conclusions and Summary





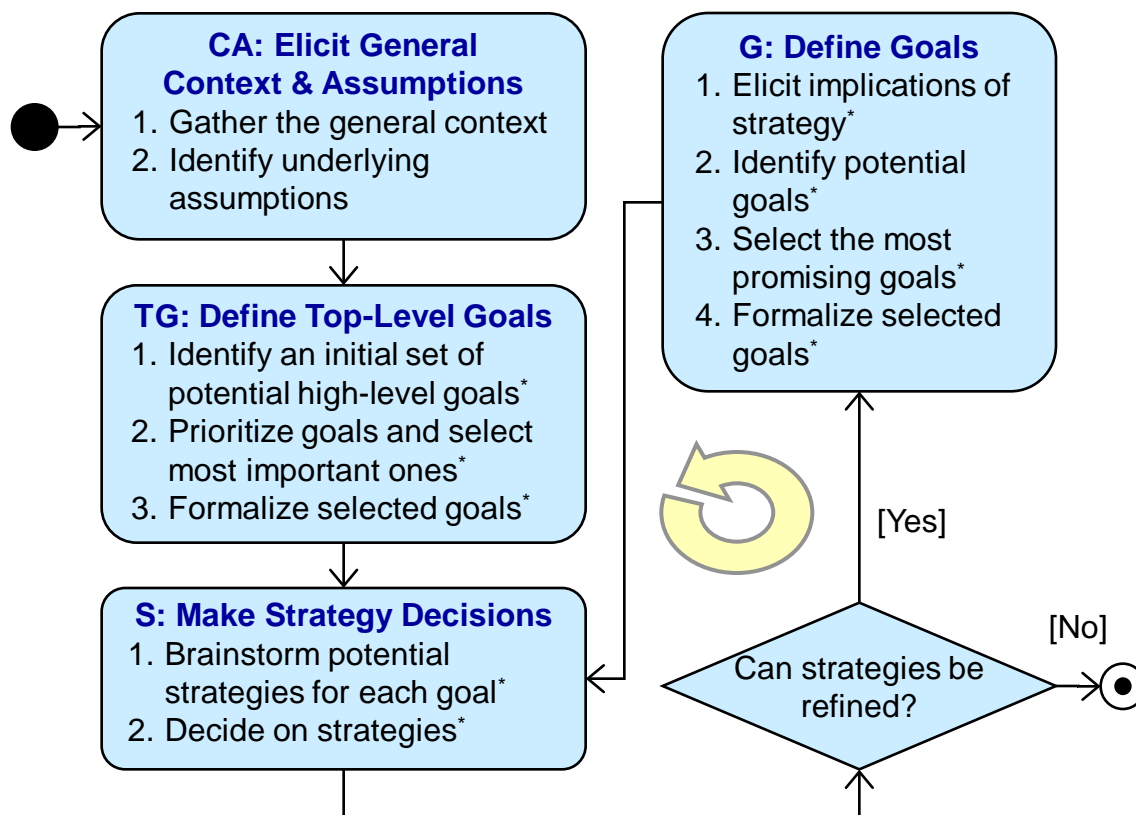
# GQM+Strategies® Life Cycle (based on the Quality Improvement Paradigm)



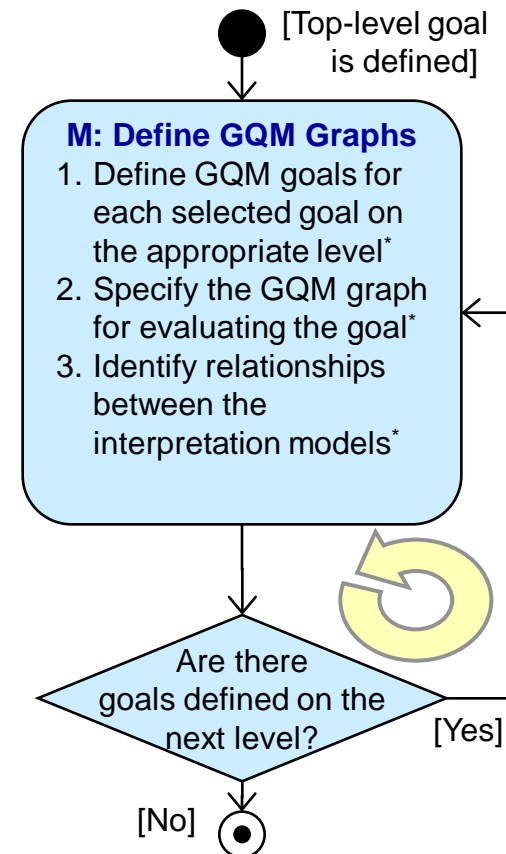


# GQM+Strategies® Grid Derivation Process (top-down, bottom-up, or a mixture)

## Goals and Strategies



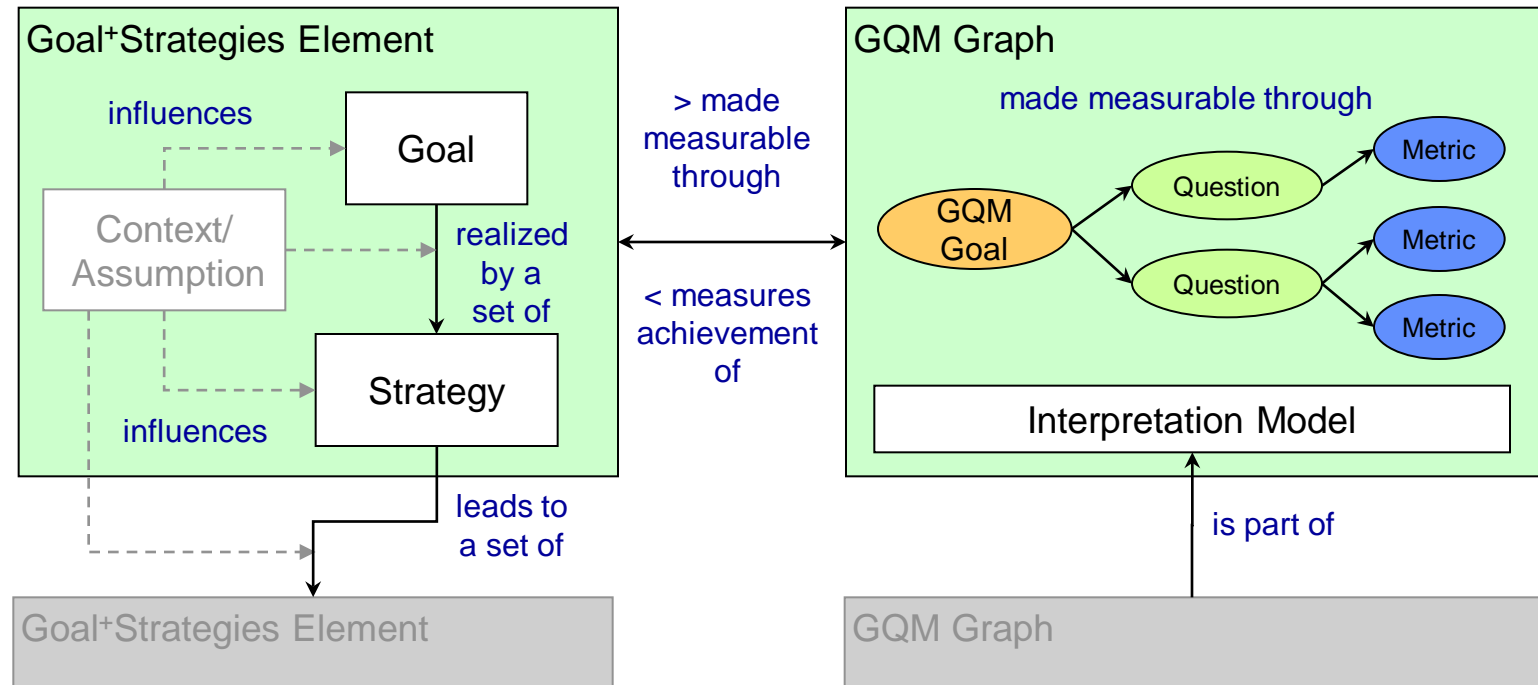
## Measurement Data



\* Document context and assumptions

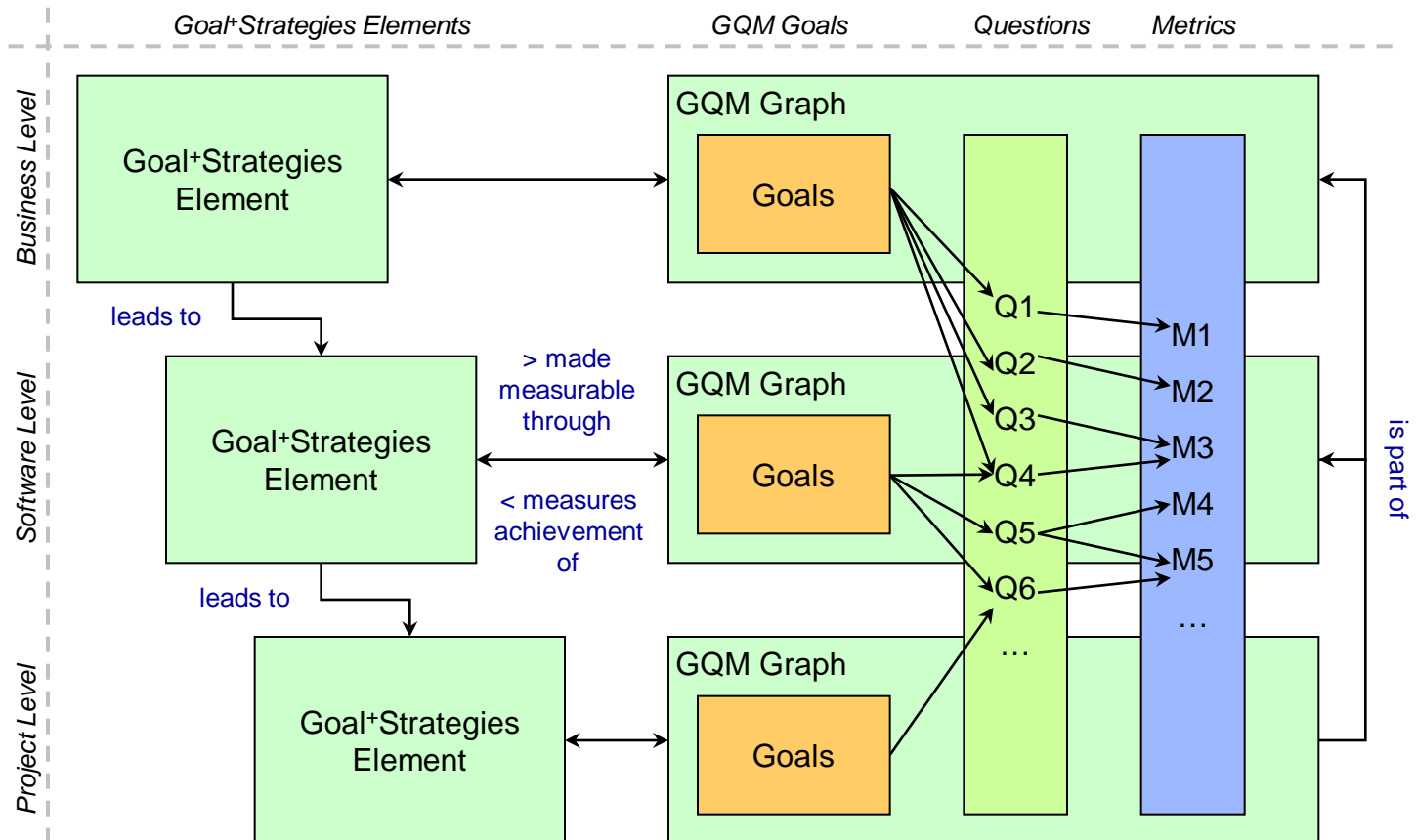


# The Goal+Strategies Meta Model





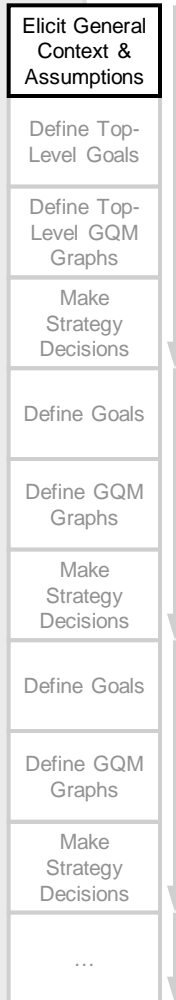
# Linking Goals at Multiple Levels in a Grid





# Basics: Elicit General Context & Assumptions

1. Gather the context for your organization (the organization for which you are setting business goals)
  - ☐ Characterize the product or service
  - ☐ Identify your existing processes, tools, etc.
  - ☐ Characterize your customers
  - ☐ Characterize income sources and business model
  - ☐ Characterize your organizational interfaces
  - ☐ Characterize the existing measurement program (goals, models, data)
2. Identify assumptions
  - ☐ What do you believe to be true but for which you have little or no empirical evidence?
  - ☐ For instance, assumptions about the technology, market, customers, your organization, workforce, etc.





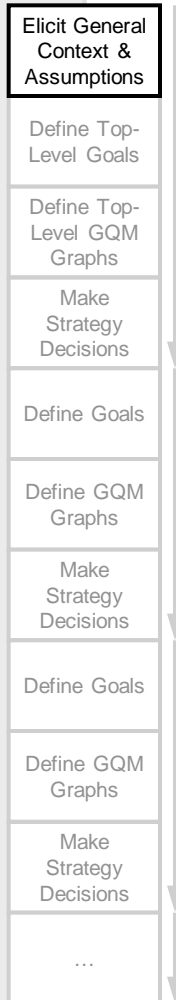
## Elicit General Context & Assumptions

### 1. Gather the context

- ☐ Company XYZ builds a class of shrink-wrapped products
- ☐ Using a standard set of processes and tools that cover the life cycle
- ☐ For the general market
- ☐ And the latest product is sold directly to customers
- ☐ This is a small organization with no contracting, i.e., they build the next version of the product themselves
- ☐ And they have some measures in place mostly to help manage products

### 2. Identify assumptions

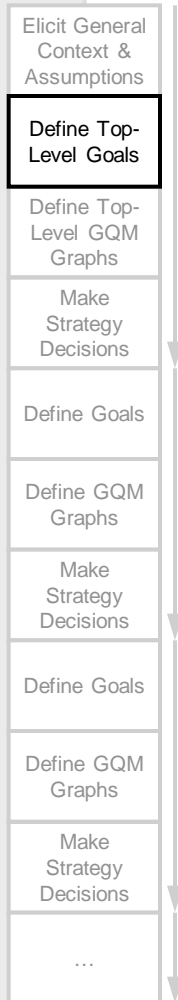
- ☐ Customer satisfaction with the product will create customer loyalty, which will cause customers to buy the next version of the product





# Basics: Define Top-Level Goals

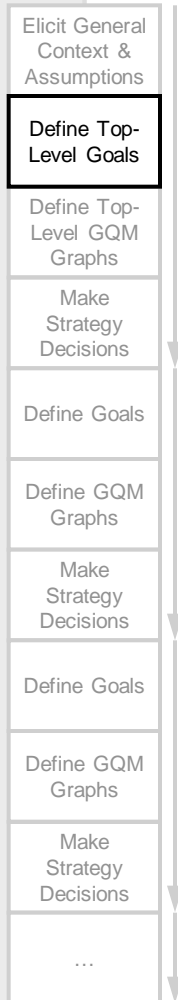
1. Identify an initial set of potential high-level goals
    - ☐ What are the organizational principles that you do not want to change, i.e., aspects of your organization you want to keep as is?
    - ☐ What are your business goals?
  2. Prioritize goals
    - ☐ Identify hierarchical relationships (sub-goals inherited by the divisions, projects, individuals)
    - ☐ Identify conflicts (complementary, conflicting, and indifferent goals)
  3. Formalize the goals
- ⇒ Document context and assumptions for all steps





## Define Top-Level Goals

- ⇒ Ask for the basic motivation (context and assumptions)
- ☐ **Context 1:** The market for our class of product is becoming highly competitive and there is a need to safeguard our place in the market.
  - ☐ **Assumption 1:** Improving customer satisfaction with each new product will lead to customer loyalty, which will help safeguard our place in the market
1. Identify an initial set of potential high-level goals
    - ☐ Increase customer satisfaction for the next product
  2. Prioritize goals
    - ☐ Only one goal was selected for Company XYZ
  3. Ask questions to formalize the business goal (using the goal template)







## Formalize Top-Level Goal

<b>Activity</b>	Increase
<b>Focus</b>	Customer satisfaction
<b>Object</b>	Product “Splash”
<b>Magnitude (degree)</b>	10% reduction in number of customer complaints
<b>Timeframe</b>	12 weeks after release
<b>Scope (context)</b>	Web Products Division, Splash Project Manager
<b>Constraints (limitations)</b>	Splash price and functionality
<b>Relationships with other goals</b>	Can conflict with development cost goals, schedule goals, ...

Elicit General Context & Assumptions

Define Top-Level Goals

Define Top-Level GQM Graphs

Make Strategy Decisions

Define Goals

Define GQM Graphs

Make Strategy Decisions

Define Goals

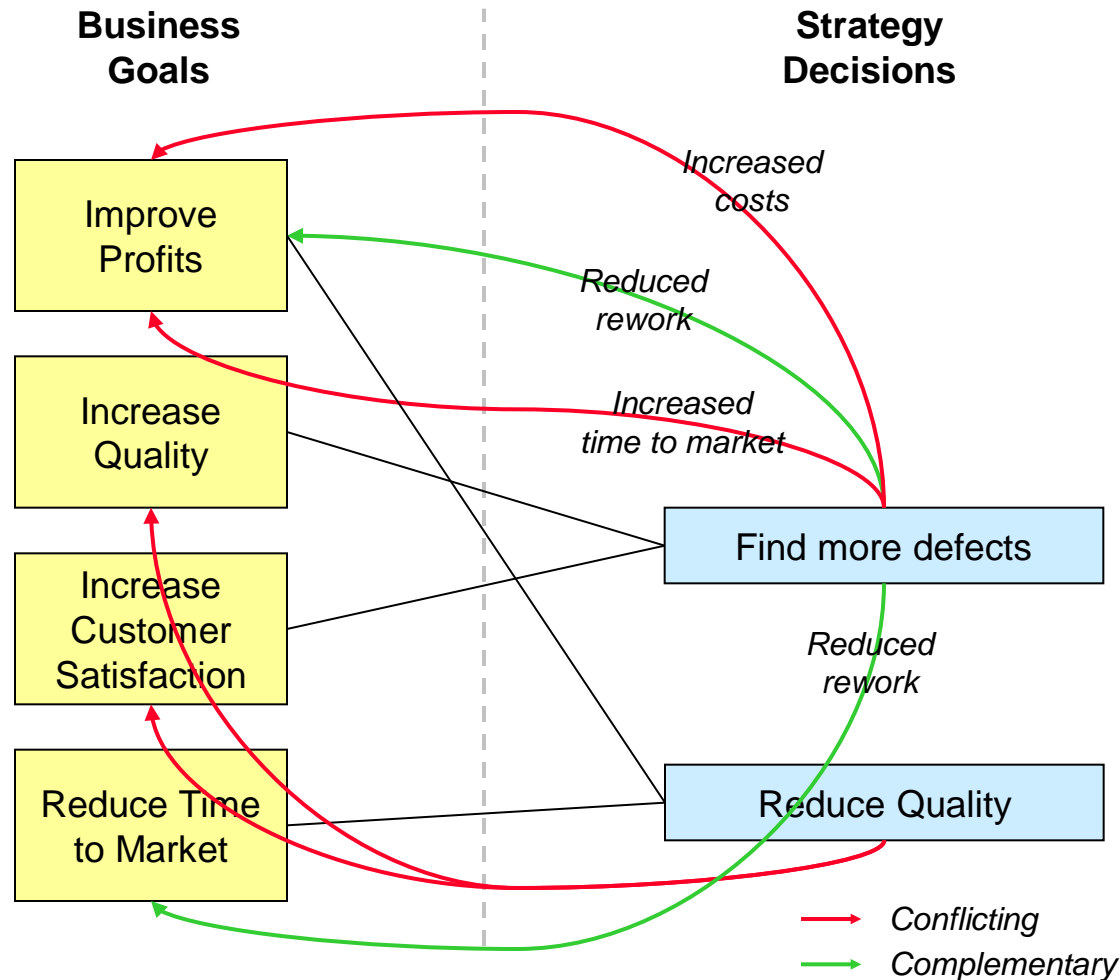
Define GQM Graphs

Make Strategy Decisions

...



## Example Relationships between Goals



### Approaches

- Explicitly model potential conflicts
- Prioritize goals
- Select strategies that minimize conflicts
- Select most important goals



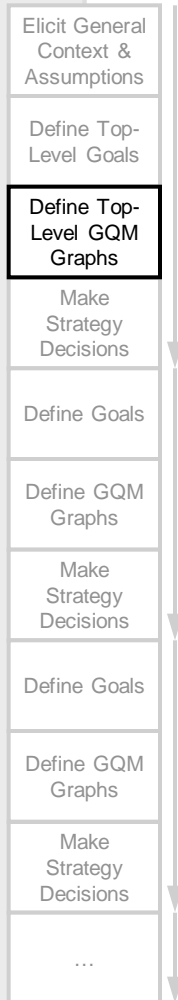
# Basics: Define Top-Level GQM Graphs

## 1. Define GQM goals

- ☐ **Object:** process, product, resource, ...
- ☐ **Purpose:** characterize, evaluate, predict, motivate, improve
- ☐ **Quality Focus:** cost, correctness, defect removal, changes, reliability, user friendliness
- ☐ **Viewpoint:** user, customer, manager, developer, corporation
- ☐ **Context:** problem factors, people factors, resource factors, process factors

## 2. Identify the GQM graph for evaluating the achievement of the goal

- ☐ Questions, measures and models
- ☐ Decision criteria in the interpretation model





## Define Top-Level GQM Graphs

### 1. Define GQM goals

- ☐ Analyze **customer complaints trend for Splash**
- ☐ for the purpose of **evaluation**
- ☐ with respect to **10% improvement over history**
- ☐ from the point of view of **quality management**
- ☐ in the context of **Web Products Division of XYZ**

### 2. Identify the GQM graph for evaluating the achievement of the business goal

- ☐ **Assumption 2:** Customer satisfaction can be measured by # of customer complaints
- ☐ **Measures and models**
  - ☐ CCS = Number of customer complaints in the first 12 weeks after release of Splash
  - ☐ CCB = Average number of customer complaints in the first 12 weeks after release of a set of baseline products
  - ☐  $CCR = CCS / CCB$
- ☐ **Decision criteria in the interpretation model**
  - ☐ If  $CCR \leq 0.9$ , the business goal is achieved



# Basics: Make Strategy Decisions

## 1. Brainstorm potential strategies

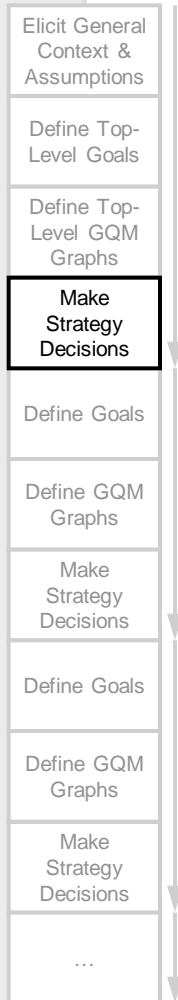
- ☐ What are the different ways that the Business Goal could possibly be achieved?

## 2. Strategy decisions

- ☐ Decide on a strategy, based on context factors, assumptions, feasibility, etc.

⇒ Document context and assumptions for all steps

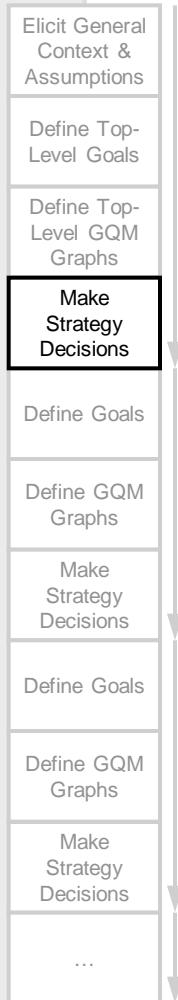
- ☐ What context factors affect the choice of strategy?
- ☐ Make explicit any assumptions that affect the choice of strategy





## Make Strategy Decisions

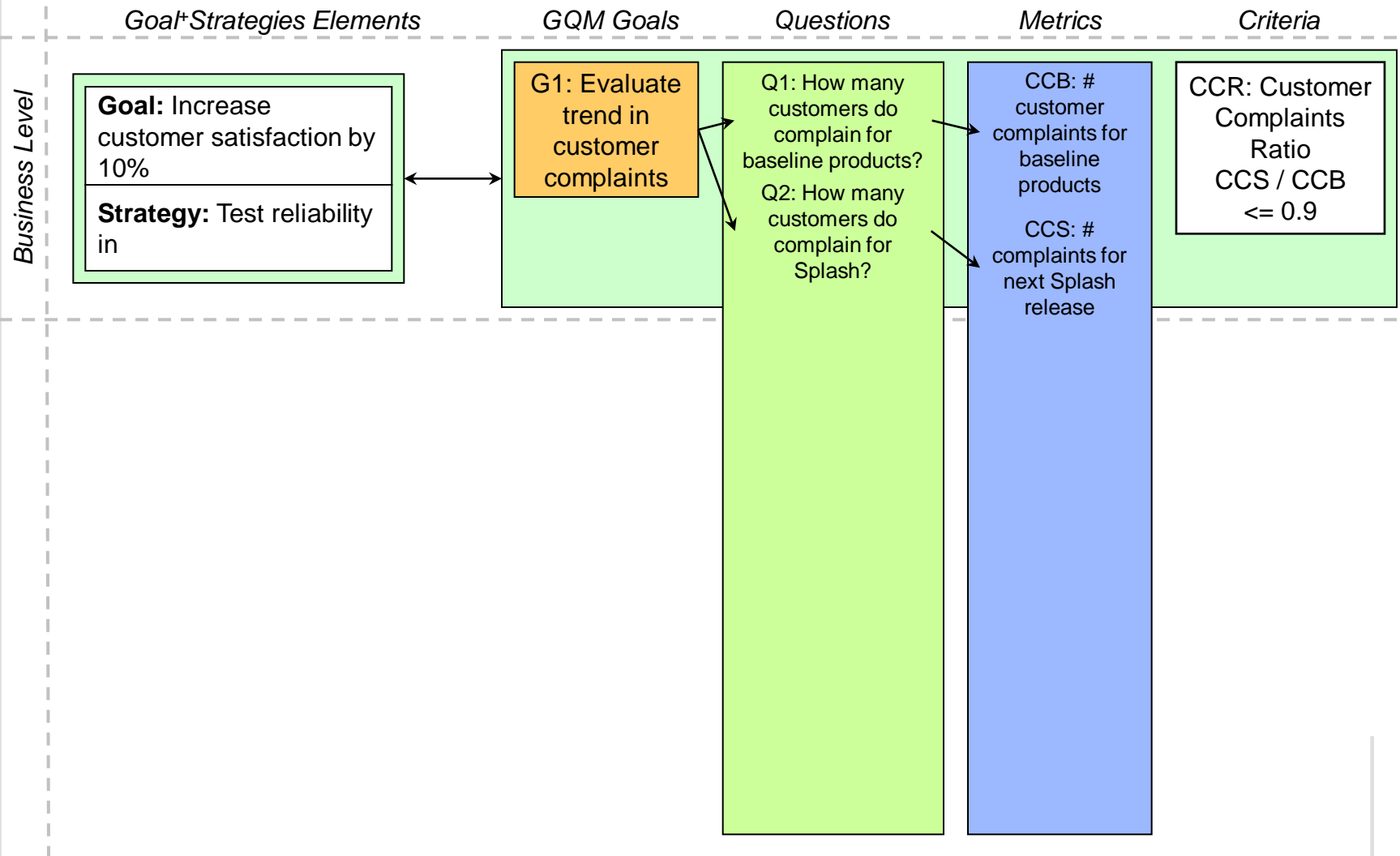
- ⇒ Document context and assumptions
  - ☐ **Assumption 3:** Many customer complaints are due to product reliability problems
- 1. Brainstorm potential strategies
  - ☐ Build reliability in (e.g., implement fewer defects)
  - ☐ Test reliability in (e.g., remove more defects)
- ⇒ Document context and assumptions
  - ☐ **Context 2:** Little control over development process (too late)
  - ☐ **Context 3:** Limited budget for process improvement
- 2. Decide on a strategy
  - ☐ Test reliability in (e.g., remove more defects)





# Example

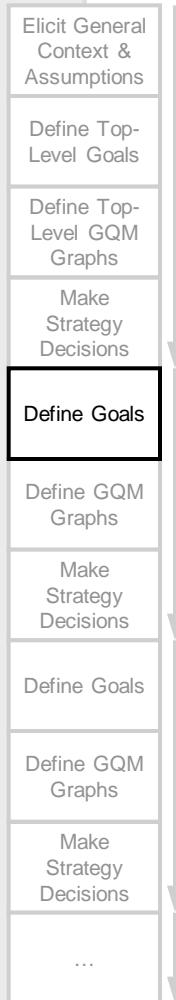
## Example Summary: GQM Part of the Grid





## Basics: Define Goals

1. Elicit the implications of the chosen strategy with respect to software development
    - ☐ In order to carry out this organizational strategy, what does the software development organization need to do?
  2. Identify potential software goals
  3. Select the most promising goal considering feasibility, cost, and benefit
  4. Ask questions to formalize the goals
    - ☐ Use the goal template
- ⇒ Document context and assumptions for all steps

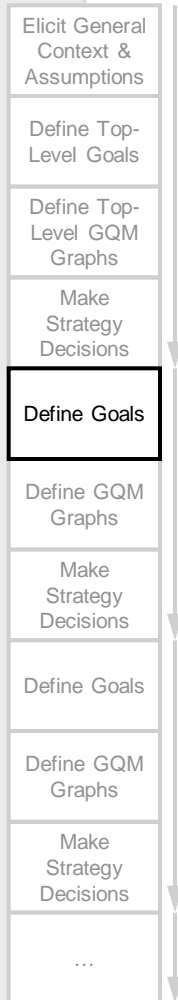






## Define Second-Level Goals (1/2)

1. Elicit the implications of the chosen strategy with respect to software development
  - ☐ To test in reliability, the software test processes must be examined
2. Identify potential software goals
  - ☐ Decrease customer-reported defects by improving
    - ☐ System test effectiveness
    - ☐ Unit test effectiveness
    - ☐ Acceptance test effectiveness





## Define Second-Level Goals (2/2)

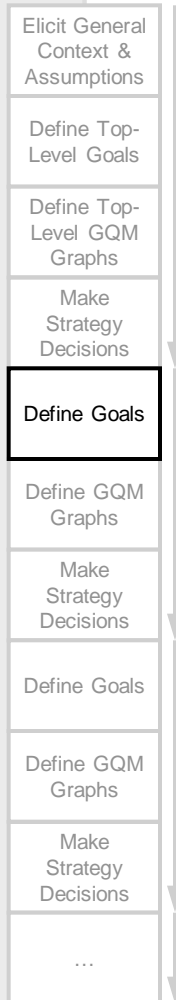
⇒ Document context and assumptions

- ☐ **Context 4:** There is a new system test process that seems appropriate for our context
- ☐ **Assumption 4:** We can decrease the total # of customer complaints by 10% by reducing customer-reported software field defects (i.e., those that slip by system test) by 20%

3. Select the most promising goal considering feasibility, cost, and benefit

- ☐ Decrease customer-reported defects by improving system test effectiveness

4. Ask questions to formalize the software goal





## Formalize Second-Level Goals

<b>Activity</b>	Decrease
<b>Focus</b>	Customer-reported software defects
<b>Object</b>	System test process for Splash
<b>Magnitude (degree)</b>	20%
<b>Timeframe</b>	12 weeks after release (might check every week)
<b>Scope (context)</b>	Web Products Division, Splash Software Manager
<b>Constraints (limitations)</b>	Development cost and functionality
<b>Relationships with other goals</b>	Can conflict with development cost goals, schedule goals, ...

Elicit General Context & Assumptions

Define Top-Level Goals

Define Top-Level GQM Graphs

Make Strategy Decisions

**Define Goals**

Define GQM Graphs

Make Strategy Decisions

Define Goals

Define GQM Graphs

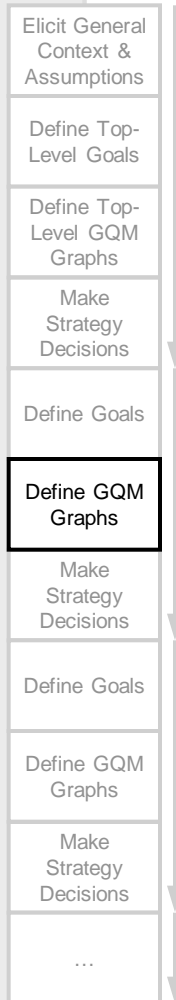
Make Strategy Decisions

...



## Basics: Define GQM Graphs

1. Define GQM goals
  2. Identify the GQM graph for evaluating the achievement of the goal
    - ☐ Measures and models
    - ☐ Decision criteria in the interpretation model
  3. Identify relationships between this interpretation model and the one for your higher-level goal
- ⇒ Document context and assumptions for all steps





## Define Second-Level GQM Graphs (1/2)

### 1. Define GQM goals

- ☐ Analyze the trend in unique customer complaints that are due to software defects
- ☐ for the purpose of evaluation
- ☐ with respect to 20% reduction when compared to prior projects
- ☐ from the point of view of quality management
- ☐ in the context of Web Products Division of XYZ

### 2. Identify the GQM graph for evaluating the achievement of the goal

- ☐ Measures and models
  - ☐ CDS = Number of unique customer complaints that are due to software defects in the first 12 weeks after release of Splash
  - ☐ CDB = Average number of unique customer complaints that are due to software defects in the first 12 weeks after release of a set of baseline products
  - ☐  $CDR = CDS / CDB$
- ☐ Decision criteria in the interpretation model
  - ☐ If  $CDR \leq 0.8$ , the goal is achieved





## Define Second-Level GQM Graphs (2/2)

3. Identify relationships between this interpretation model and the one for your higher-level (business) goal
  - ☐ **If**  $CCR \leq 0.9$  (i.e., the number of customer complaints is reduced by 10%)
  - ☐ **Then** we have achieved our business goal
  - ☐ **Else**
    - ☐ **If**  $CDR \leq 0.8$  (i.e., the number of unique customer complaints due to software defects is reduced by 20%)
    - ☐ **Then** Assumption 4 is wrong
      - ☐ **Check Assumption 4:** We can decrease the total # of customer complaints by 10% by reducing customer-reported software field defects by 20%
    - ☐ **Else** reconsider the “testing in” strategy

Elicit General Context & Assumptions

Define Top-Level Goals

Define Top-Level GQM Graphs

Make Strategy Decisions

Define Goals

Define GQM Graphs

Make Strategy Decisions

Define Goals

Define GQM Graphs

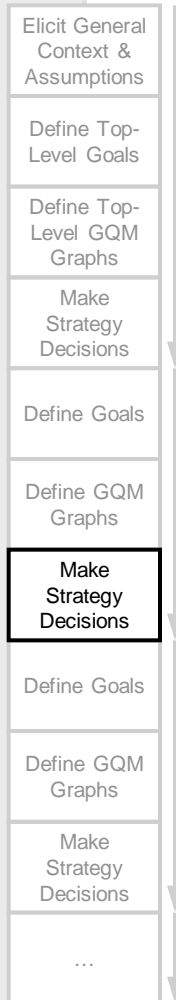
Make Strategy Decisions

...



## Make Second-Level Strategy Decisions

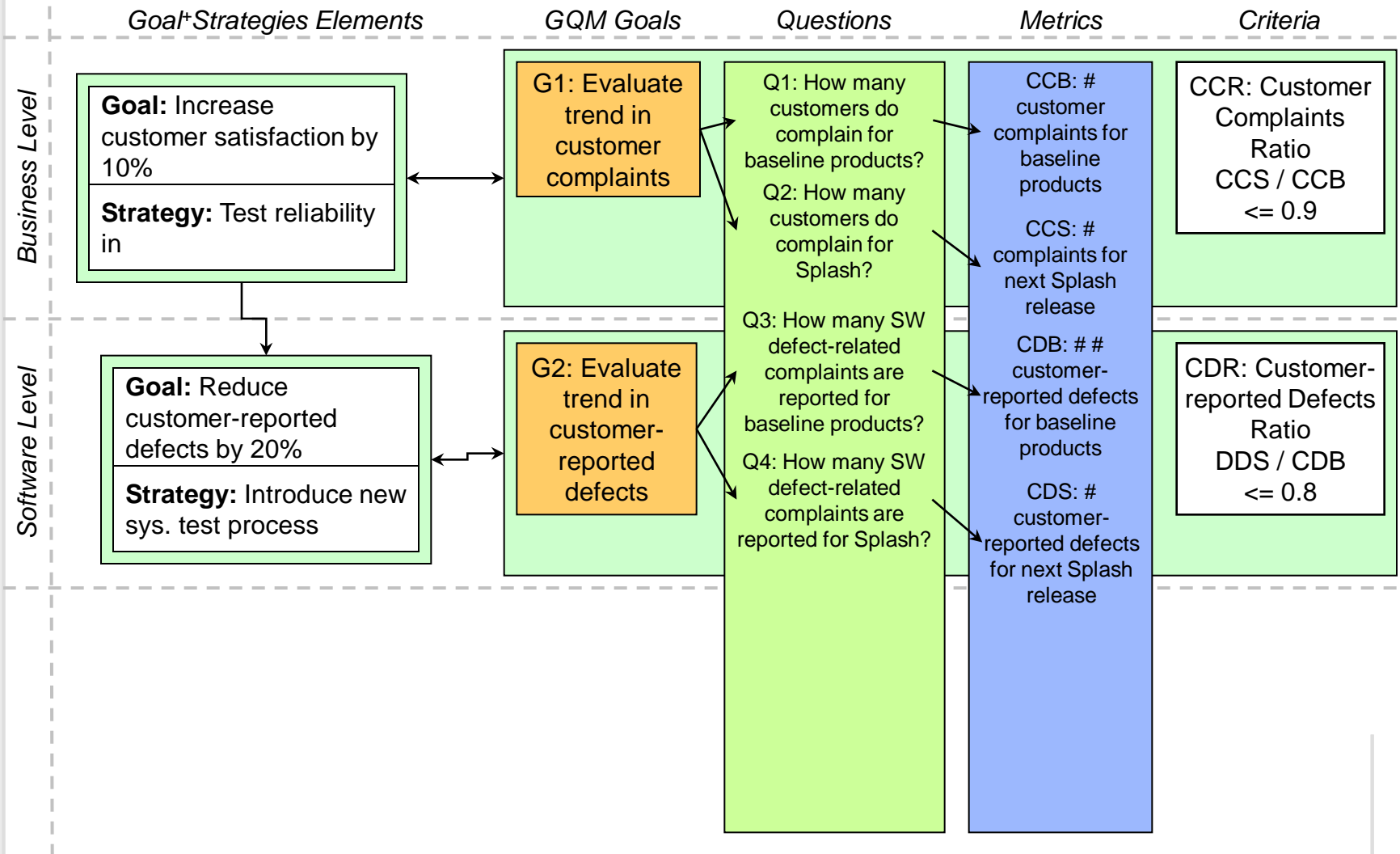
1. Brainstorm potential strategies
  - ☐ Because of **Context 4** (“There is a new system test process that seems appropriate for our context”), the one and only strategy is to introduce a new system test process
2. Decide on a strategy
  - ☐ Introduce new system test process





# Example

## Example Summary: GQM Part of the Grid

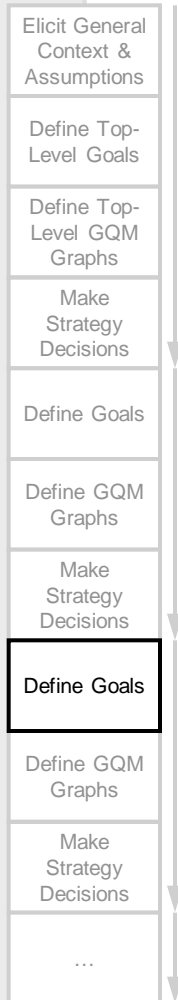






## Define Third-Level Goals

- ⇒ Document context and assumptions
- **Assumption 5:** Reducing slippage by 20% reduces reported defects by 20%
1. Elicit the implications of the chosen strategy with respect to the next level
    - Apply the new system test method to see if it reduces defect slippage by at least 20% and generates the necessary improvement regarding customer complaints
  2. Identify potential goals
    - Reduce system test defect slippage by 20%
  3. Select the most promising goals considering feasibility, cost, and benefit
    - Reduce system test defect slippage by 20%
  4. Formalize the goals





## Formalize Third-Level Goals

<b>Activity</b>	Decrease
<b>Focus</b>	Defect slippage
<b>Object</b>	New system test process for Splash
<b>Magnitude (degree)</b>	20%
<b>Timeframe</b>	12 weeks after release (might check every week)
<b>Scope (context)</b>	Web Products Division, Splash Software Manager
<b>Constraints (limitations)</b>	Development cost and functionality
<b>Relationships with other goals</b>	Can conflict with development cost goals, schedule goals, ...

Elicit General Context & Assumptions

Define Top-Level Goals

Define Top-Level GQM Graphs

Make Strategy Decisions

Define Goals

Define GQM Graphs

Make Strategy Decisions

Define Goals

Define GQM Graphs

Make Strategy Decisions

...



## Define Third-Level GQM Graphs (1/2)

### 1. Define GQM goals

- ☐ Analyze the system test process for Splash
- ☐ for the purpose of evaluation
- ☐ with respect to 20% defect slippage compared to prior projects
- ☐ from the point of view of quality management
- ☐ in the context of Web Products Division of XYZ

### 2. Identify the GQM graph for evaluating the achievement of the goal

- ☐ Measures and models
  - ☐ DP = ratio of defects found in system test to those found after system test (customer reported + found in acceptance test) on this project
  - ☐ DB = the ratio of defects found in system test to those found after system test in baseline set of projects
  - ☐  $DSR = DP / DB$
- ☐ Decision criteria in the interpretation model
  - ☐ If  $DSR \geq 1.2$ , there is at least a 20 % improvement
  - ☐ If  $1 \leq DSR < 1.2$ , method is better than history but not good enough

Elicit General  
Context &  
Assumptions

Define Top-  
Level Goals

Define Top-  
Level GQM  
Graphs

Make  
Strategy  
Decisions

Define Goals

Define GQM  
Graphs

Make  
Strategy  
Decisions

Define Goals

Define GQM  
Graphs

Make  
Strategy  
Decisions

...



## Define Third-Level GQM Graphs (2/2)

3. Identify relationships between this interpretation model and the one for your higher-level (software) goal
  - ☐ **If** the customer-reported defects are reduced by at least 20%,
  - ☐ **Then** we have achieved our software goal
  - ☐ **Else**
    - ☐ **If** the defect slippage from system test is reduced by at least 20%
    - ☐ **Then** Assumption 5 or 6 is wrong
      - ☐ **Check Assumption 5:** Reducing slippage by 20% reduces reported defects by 20%
      - ☐ **Check Assumption 6:** The projects that form the baseline are relevant to the current project
    - ☐ **Else** reconsider the software strategy
- There are more GQM goals (not discussed in this example) on this level that could be followed depending on whether historical data on defect slippage exists.

Elicit General  
Context &  
Assumptions

Define Top-  
Level Goals

Define Top-  
Level GQM  
Graphs

Make  
Strategy  
Decisions

Define Goals

Define GQM  
Graphs

Make  
Strategy  
Decisions

Define Goals

Define GQM  
Graphs

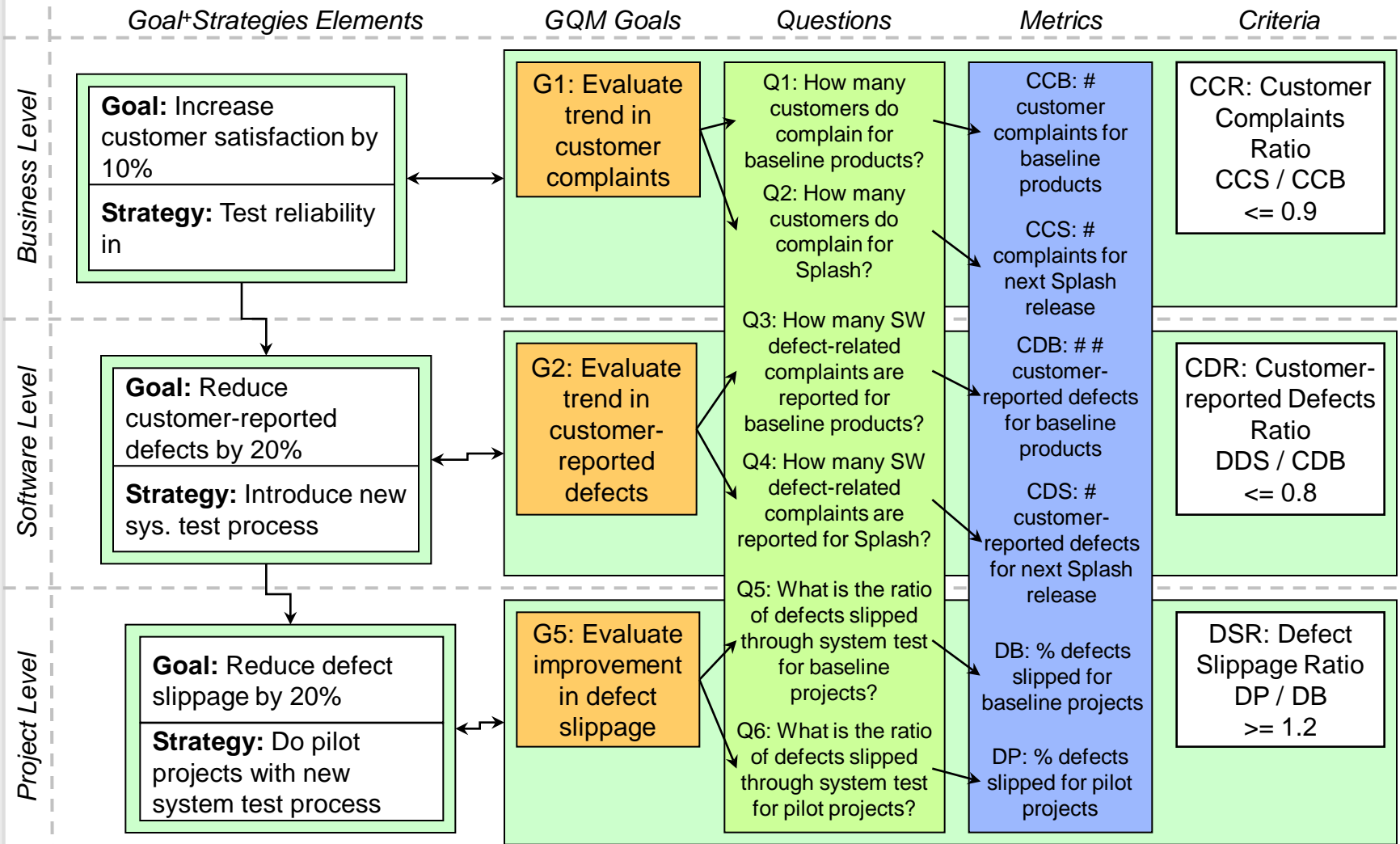
Make  
Strategy  
Decisions

...



# Example

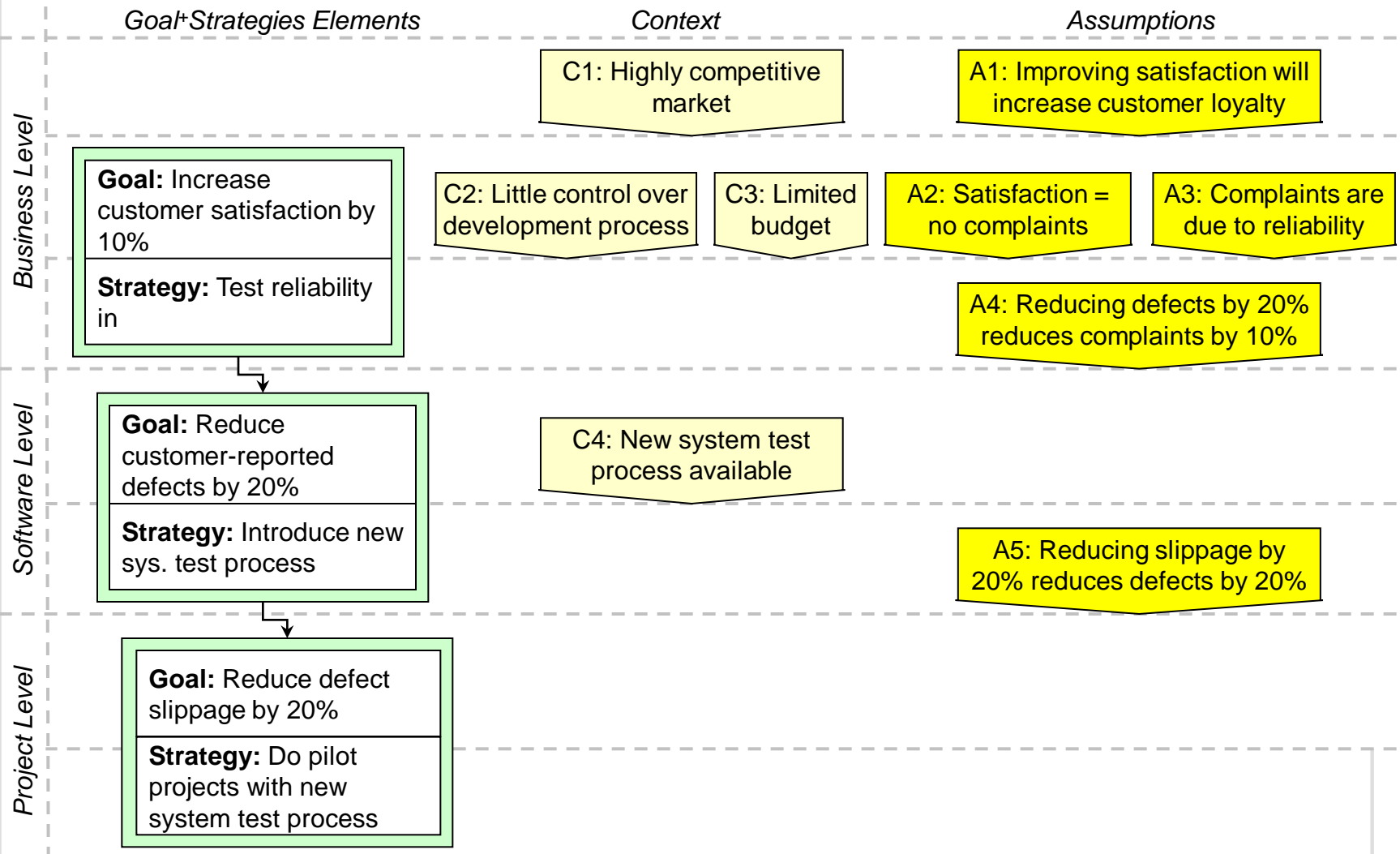
## Example Summary: GQM Part of the Grid





# Example

## Example Summary: Context and Assumptions of the Grid





# What happens next?

## ■ Getting Commitment

- ☐ Splash Software Manager goes to the Web Products Manager and asks for the resources;
- ☐ to justify this request, the Splash Software Manager can point to a clear link between the resources needed and the business goal they support

## ■ Detailed Planning

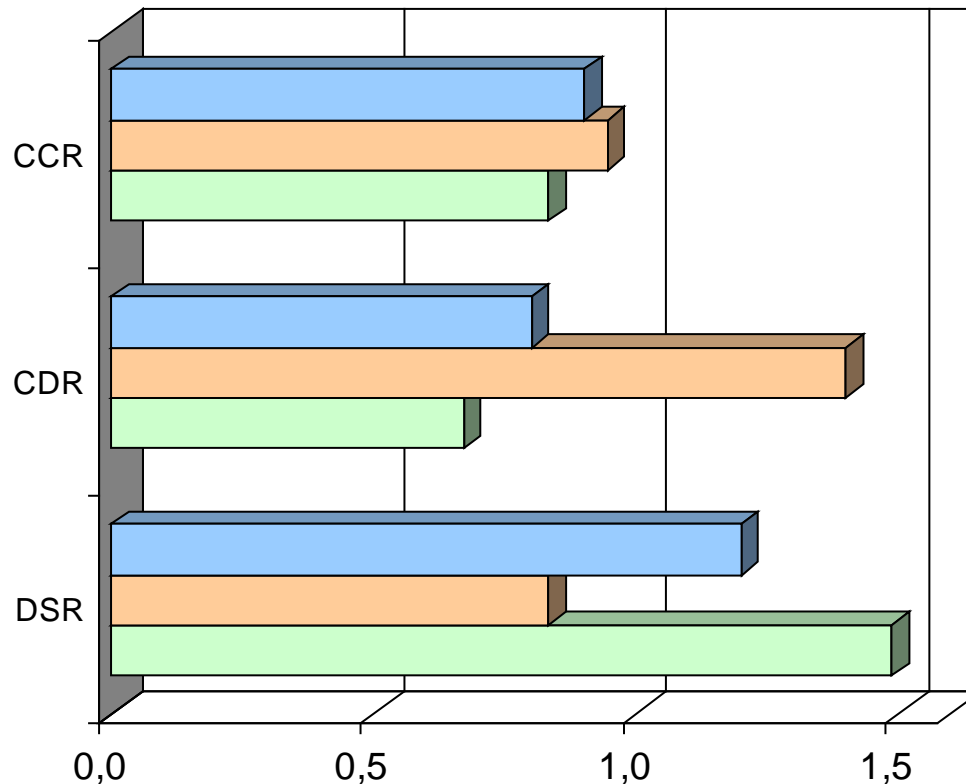
- ☐ Splash Software Manager creates detailed plan of activities (the steps of the strategy), how the resulting information needs to be passed back up the line (the strategy), and who is interested

## ■ Execution and Monitoring

- ☐ Web Products Manager gets the information needed to show how customer satisfaction (the original success goal) is being addressed, and whether it's effective or not



## Example Data Visualizations of Decision Criteria



- $CCR \leq 0.9$   
 $\Rightarrow$  at least 10% improvement
- $CDR \leq 0.8$   
 $\Rightarrow$  at least 20% improvement
- $DSR \geq 1.2$   
 $\Rightarrow$  at least 20% improvement
- Outcome 2 meets all three criteria



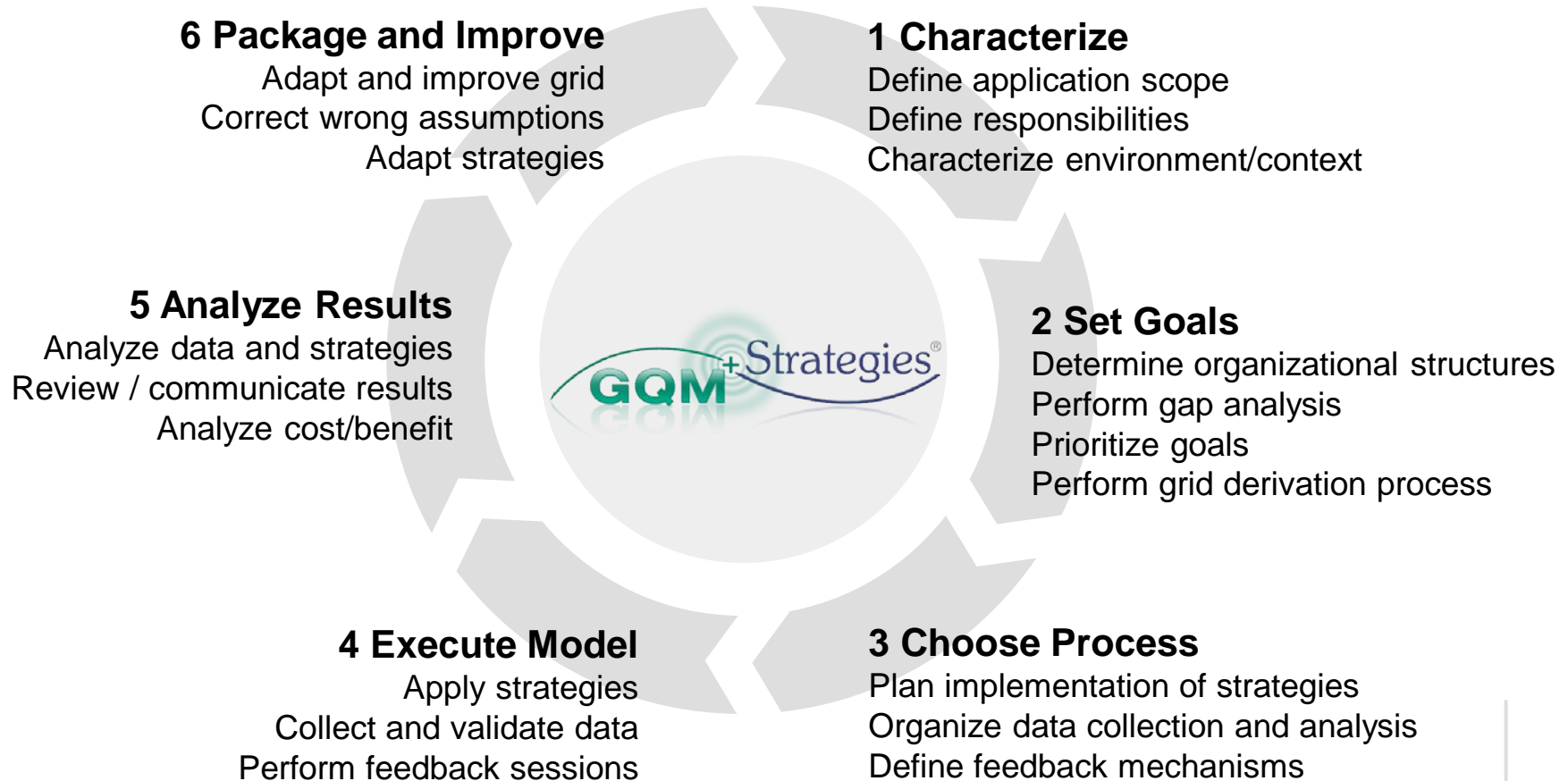


## Measurement-based Alignment of IT Strategies and Business Goals

- Introduction and Motivation
- Goal-oriented Software Measurement
- GQM+Strategies® Method
- **Conclusions and Summary**



# **GQM+Strategies® Life Cycle** (based on the Quality Improvement Paradigm)





# Strategic Planning Addresses Various Stakeholders

## ■ Business customers

- ☐ Demonstrate understanding of their business and their goals
- ☐ Set expectations - what can (reasonably) be expected
- ☐ Clarify the rules under which IT operates - governance, prioritization, allocation of resources

## ■ Executive management

- ☐ Show alignment with corporate priorities
- ☐ Communicate rationale behind IT budget and longer-term investment needs
- ☐ Describe governance controls

## ■ IT people

- ☐ Clarify expectations - what needs to be done for being successful
- ☐ Provide “room to maneuver” - how to balance new and conflicting demands on IT
- ☐ Justify investments - applications and technology portfolio, and IT skills and processes

Source: Forrester Research, Inc.



## GQM+Strategies® Contributions

- Clarify and harmonize goals, strategies, and measurement data
- Illustrate value of IT/software in terms of larger business objectives
- Improve communication of goals throughout an organization
- Support transparent decision making

Business	Domain	Application
European telecommunications company	Telecommunications	Drive strategic improvement programs
European automotive supplier	Automotive	Support Measurement and Analysis
European network testing company	Telecommunications	Modernizing existing product suite
International software company	Embedded systems used in telecommunications	Increase the visibility of strategic decisions
Asian insurance company	Information systems	Address new business domain
Asian systems engineering organization	Safety-critical software for aerospace domain	Enhance supplier collaboration
International gas and oil company	Information systems	Alignment and value of IT



## Success Factors and Recommendations

- Get Support
  - Missing management commitment
- Get Acceptance
  - Project manager and project members are difficult to convince regarding the benefits of measurement
  - Fear of misusing measurement data, e.g., for being assessed
  - Getting data from external organizations (e.g., suppliers)
- Avoid Manipulations
  - Analysis results may be manipulated in order to make the results look better
- Steer Effectiveness
  - Measurement program is not as effective as expected
- Control Costs
  - Measurement costs too much overhead



## Conclusions

- GQM+Strategies®
  - ☐ Provides **justification and accountability** at all levels
  - ☐ Provides a **clear plan of action**
  - ☐ Provides **guidance** not just for planning, but also for analyzing and rolling up the resulting data to the people who need to make decisions
  - ☐ Provides support for **defining and sustaining the strategic measurement** and management process
    - ☐ Creating the right organizational structure
    - ☐ Getting feedback to projects in a timely fashion
    - ☐ Maintaining commitment within all organizational levels

# Thanks for your attention!