Problem Solving Grand Slam: 7 Steps to Master

Any Industry, Any Problem, Every Project

Summer 2013
The 7-Steps to master problem solving

1. Define problem
   - Problem statements should commence with a question or a firm hypothesis
   - Be specific, actionable and focus on what the decision maker needs to move forward

2. Build Issue Tree
   - Break a problem into component parts so that problems can be divided and allocated
   - The parts should be MECE
   - Do it as a team, share with Experts and client to get input and alignment

3. Write Storyline and Ghost pack
   - Write your best version of the answer to the project – STORYLINE
   - Develop ghost pack that supports this Storyline
   - Prioritise key issues and eliminate non-essential issues

4. Develop Workplan
   - A workplan is not just a Gantt chart, it must state the:
     - Issue
     - Hypotheses
     - Analysis (frameworks, process etc)
     - Sources
     - Who is responsible
     - Timing
     - End product
     - Review cycles

5. Gather Data and Analyse Critically
   - Be hypothesis and end-product oriented
   - Start with order of magnitude
   - “Porpoise” between hypothesis and data
   - Keep the analyses simple
   - Be flexible in the face of new data
   - Be creative

6. Synthesize and So What
   - Drive to end of week answer
   - Tool #1: Employ Situation- Observation- Implication framework
   - Tool #2: Follow pyramid storyline structure
   - Pass the 3 So Whats

7. Build Commitment
   - Follow a Communication Plan (T-5, T-2, etc)
   - Validate, validate, validate – close to your enemies
   - Have the extra page in your back pocket

Engage Clients, Stakeholders & Experts
Why hypothesis-based approach?

Deductive Analysis Approach (found in most client organisations)

Extensive Data Collection

- Data Analysis

Conclusions & Recommendations

Inductive Analysis Approach (used in Phase 1)

Hypotheses

Analysis

Results T/F

Conclusions & Recommendations

Areas for improvements could be:

- Use a range of specific diagnostics to test hypotheses

Speed, Effort, Risk & Stakeholder Engagement

7-step approach to problem solving
Define problem

X bank is losing money on broker introduced business and needs a plan to fix it

How should X bank restructure its broker relationships to make better returns

What set and sequence of initiatives should X bank implement to cement its more profitable broker relationships and to change the laws of remunerating the larger number of unprofitable relationships?

Evaluation

Statement of fact

Not disputable and too general

Specific, actionable
Gaining an accurate understanding of the problem is the first priority

'So, Andre! . . . the king wants to know how you're coming with St George and the Dragon'
Clear documentation of the problem is equally important.
Why use logic or issue trees?

1. To break a problem into component parts so that:
   - Problem solving work can be divided into intellectually manageable pieces
   - Priorities can be set between the parts
   - Responsibilities can be allocated to individuals

2. To ensure that the integrity of the problem solving is maintained
   - Solving the parts will really solve the problem
   - The parts are mutually exclusive and collectively exhaustive (ie. No overlaps, no gaps)

3. To build a common understanding within the team of the problem solving framework

4. To help focus use of organising frameworks and theories
## Three types of logic/issue tree

<table>
<thead>
<tr>
<th>Type of Tree</th>
<th>Description</th>
<th>Elements in Splits</th>
<th>When to Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>Deductive</td>
<td>Starts with problem definition and divides it into components</td>
<td>Actions, assertions, questions, categories</td>
<td>• Early on, when you don’t know much OR • When mathematical completeness is important (eg. ROIC Trees)</td>
</tr>
<tr>
<td>Hypothesis Driven</td>
<td>Postulates a solution hypothesis and develops a necessary and sufficient rationale to validate or disprove it</td>
<td>Reasons / Questions</td>
<td>• At any point in the process</td>
</tr>
<tr>
<td>Issue Map</td>
<td>Phrases key issues so that they can be answered yes or no, and sequence them in a logical order showing the dependent action</td>
<td>Questions</td>
<td>• Use issue maps to frame options, usually later in the process</td>
</tr>
</tbody>
</table>
There are many ways to disaggregate a problem

- Vertical Characteristics
  - Suitable bricks
  - Properly mixed mortar
  - Skilled labour
  - Good weather

- Horizontal Characteristics
  - Criteria

7-step approach to problem solving:
1. Define problem
2. Build Issue Tree
3. Prioritise & Write Ghostpack
4. Develop Workplan
5. Gather Data & Analyse Critically
6. Synthesize & Iterate
7. Build Commitment
# Logic tree tips

## Principle
- Use your whole team, not lone wolf approach
- If stuck, try building them backward (from the ‘twigs’) in addition to forward (from the ‘trunk’)
- Try multiple trees and constantly update and revise
- Use frameworks to guide your questions

## Why
- Rough and tumble hypothesis sessions tap everyone’s creativity, tend to get better answers and are fun
- It is sometimes easier to think up sub-issues and analysis and then to group them, than to work linearly
- Different trees provide new perspectives on the problem
- Leverages previous experience

## Rules to follow

### Hypotheses are statements that:
- identify a client organisation's weakness, gap or opportunity
- we believe are likely to be true
- are (sometimes) based on our knowledge of the client’s industry
- can be validated or refuted by analysis during the Phase 1
- if validated, are likely to represent a significant business case element
- can be translated into project streams for Phase 2

## How to get our thinking going

### What is the issue?
- What is not working?
- Where is the process broken?

### What do you think is causing the problem?
- What are the key drivers of the issue?

### What is the effect?
- Why do we care?
- Where is the opportunity?
Create a ghost pack

<table>
<thead>
<tr>
<th>Headline (top line on each powerpoint slide)</th>
<th>Content (of each slide)</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Payments Transformation program is delivering to its original plan</td>
<td>Diagram showing progress of work including Hub I and vendor gap assessment (June?)</td>
</tr>
<tr>
<td>Results from the vendor gap assessment and initial architecture design suggests the cost to implement would exceed original estimate</td>
<td>Gap assessment (June?)</td>
</tr>
<tr>
<td>We intend to work harder to make the Fundtech/SOA solution fit the original budget...</td>
<td>TIBCO diagram (David?)</td>
</tr>
<tr>
<td>To date, the business justification for payments transformation is centered on “unraveling the spaghetti to deliver systems robustness”</td>
<td>Analysis showing “how” at a conceptual level (David?)</td>
</tr>
<tr>
<td>We haven’t yet told you about the other half of the story...the ambition of delivering a truly transformational program for customers, shareholders and employees</td>
<td>Perhaps a portfolio approach it IT assets?</td>
</tr>
<tr>
<td>Our vision is to be ahead of the game by 2017</td>
<td>Use the old spaghetti diagram turning into the SOA construct in 2 stages (Ed?)</td>
</tr>
<tr>
<td>...deliver a richer set of functional capabilities to enrich customer experience, increase value to them, thereby earn heir “stickiness”</td>
<td>Schematic showing what we mean by more value to the three stakeholder groups (Ed coordinate input from Paul, James and Roger?)</td>
</tr>
<tr>
<td>...increase our capability to serve the specific needs of different segments</td>
<td>Framework to rate Westpac against other banks (now and in the future) – James/Ed?</td>
</tr>
<tr>
<td>...reduce total cost of ownership by reducing errors and adopting a portfolio approach to systems replacement</td>
<td>Current vs future (David?/Phil/Ed?)</td>
</tr>
<tr>
<td>...align the scale of IT against customer segment need</td>
<td>Current vs future (David?)/Phil/Ed?</td>
</tr>
<tr>
<td>...locate functions in the right geographies (in terms of cost and capability</td>
<td>Current vs future (Ed)?</td>
</tr>
<tr>
<td>...compliance?</td>
<td>Current vs future (June?)</td>
</tr>
<tr>
<td>...systems robustness?</td>
<td>Current vs future (David?)</td>
</tr>
<tr>
<td>...straight through processing, tracking capability?</td>
<td>Current vs future (David?)</td>
</tr>
<tr>
<td>...SOA</td>
<td>David?</td>
</tr>
<tr>
<td>Our strategy is to better utilize Westpac investment budget</td>
<td>Current vs future (Roger/Phil/Ed?)</td>
</tr>
<tr>
<td>Our roadmap to get there</td>
<td>$50m used to used on technology</td>
</tr>
</tbody>
</table>

Why create ghost pack when you haven’t even started looking at the data?

1. Enables you/your team to be output oriented
2. Forces you to think about the analyses you could need to conduct, and the data you need to gather (avoiding risk of trying to “boil-the-ocean” or be buried in too much data)
3. Brings greater alignment between the analyses you plan to conduct and the client’s expectations
4. Enables you to assign slides to team members
5. Helps identifies gap in logic and storyline early
# Build a specific workplan

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Define problem</td>
</tr>
<tr>
<td>2</td>
<td>Prioritise &amp; Write Ghostpack</td>
</tr>
<tr>
<td>3</td>
<td>Develop Workplan</td>
</tr>
<tr>
<td>4</td>
<td>Gather Data &amp; Analyse Critically</td>
</tr>
<tr>
<td>5</td>
<td>Synthesize &amp; Iterate</td>
</tr>
<tr>
<td>6</td>
<td>Build Commitment</td>
</tr>
<tr>
<td>7</td>
<td>Weekly cycle</td>
</tr>
</tbody>
</table>

## Definition

<table>
<thead>
<tr>
<th>Issue</th>
<th>Hypothesis</th>
<th>Analysis</th>
<th>Source</th>
<th>Responsibility/ Timing</th>
<th>End Product</th>
</tr>
</thead>
<tbody>
<tr>
<td>• May start with end points from logic tree. An issue varies from an important question to an unresolved question. It is phrased so that it can be answered yes or no and on which a specific action depends</td>
<td>• The hypothesis is a statement of the likely resolution of the issue. It includes the reason for answering yes or no.</td>
<td>• Which ‘models’ will be explored in order to prove or disprove the hypothesis, and hence resolve the issue</td>
<td>• The source identifies the likely location or means of obtaining data to undertake analysis.</td>
<td>• Responsibility identifies the person who will obtain the data and undertake the analysis.</td>
<td>• The end product is a statement of the output from the analysis.</td>
</tr>
</tbody>
</table>

## Action

<table>
<thead>
<tr>
<th>Issue</th>
<th>Hypothesis</th>
<th>Analysis</th>
<th>Source</th>
<th>Responsibility/ Timing</th>
<th>End Product</th>
</tr>
</thead>
</table>
| • Make sure each issue is stated in detail  
• Define sub-issues where necessary | • List all hypotheses use all ideas.  
• Discuss with team members, refine hypothesis, readjust priorities for analysis | • Identify decision making.  
• Determine extent of analysis required. | • Identify readily available data.  
• Decide on methodology. | • Decide who will help collect the data and do analysis.  
• Decide on time frame, with milestones. | • Draw ghost packs.  
• Develop story line. |
A little disciplined, early workplanning can go a long way in problem solving
### Principles and Comment

<table>
<thead>
<tr>
<th>Principles</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Be hypothesis-driven, and end-products oriented</td>
<td>• Don’t just “run the numbers” – ask “what question am I trying to answer?”</td>
</tr>
<tr>
<td>• “Porpoise” frequently between hypothesis and data</td>
<td>• Don’t chase your tail</td>
</tr>
<tr>
<td>• Keep the analyses as simple as you can</td>
<td>• Be suspicious of huge linear programs and their ilk</td>
</tr>
<tr>
<td>• Do order of magnitude estimates before you start detailed analyses</td>
<td>• Keep your eye on the forest</td>
</tr>
<tr>
<td>• Use 80/20 and back-of-envelope thinking</td>
<td>• Beware of “polishing dirt”</td>
</tr>
<tr>
<td>• Use experts as data sources</td>
<td>• Often give clearer direction than “library data”</td>
</tr>
<tr>
<td>• Be flexible in the face of new data</td>
<td>• Remember your hypothesis is there to disprove</td>
</tr>
<tr>
<td>• Share good ideas with the team</td>
<td>• (and test your thinking)</td>
</tr>
<tr>
<td>• Anticipate obstacles</td>
<td>• Keep one eye out in front of you</td>
</tr>
<tr>
<td>• Don’t be afraid to be creative</td>
<td>• Look for breakthrough thinking</td>
</tr>
</tbody>
</table>

**ASHEI**

**Gather data and analyse critically**

**Weekly cycle**

1. Define problem
2. Build Issue Tree
3. Prioritise & Write Ghostpack
4. Develop Workplan
5. Gather Data & Analyse Critically
6. Synthesize & Iterate
7. Build Commitment

- **Define problem**
- **Build Issue Tree**
- **Prioritise & Write Ghostpack**
- **Develop Workplan**
- **Gather Data & Analyse Critically**
- **Synthesize & Iterate**
- **Build Commitment**

**7-step approach to problem solving**

1. **Define problem**
2. **Build Issue Tree**
3. **Prioritise & Write Ghostpack**
4. **Develop Workplan**
5. **Gather Data & Analyse Critically**
6. **Synthesize & Iterate**
7. **Build Commitment**
Where possible, choose simple approaches...

...and avoid complicated, indirect, or inferential methods

“I say fifty, maybe a hundred horses . . . What you say, Red Eagle?”
Aim for “sufficient precision” – often perfection

“Well, shoot... I can never tell whether these things are done or not”

Be sure to fully leverage the experience of others...

“Say... look what they’re doing”
... and always look for experts to help guide your analyses

Check to be sure conclusions match up with all the facts...

'Freeze! ... okay, now ... who's the brains of this outfit?'

'And the murderer is ... the butler! Yes, the butler ... who I'm convinced, first gored the Colonel to death before trampling him to smithereens'
... and be prepared to revise your hypothesis as evidence accumulates

Always look for creative approaches...

“Say... what's a mountain goat doing way up here in a cloud bank?”
Synthesize findings and build argument and recommendation

So what…

- Throughout your analysis, you should be asking yourself “so what” three times:
  1. **So what** does this mean?
  2. **So what** does this mean for the client?
  3. **So what** does the client do about this?

Answering these questions will lead you to useful and practical recommendations.

Use the pyramid logic structure to present recommendations and supporting arguments. The model is:

```
Situation  |  Situation
Complication  |  Opportunity
Question  |  Recommendation
Recommendation
```
Forming recommendations

- Ensure recommendations are thought through in terms of implications. Keep asking yourself, “If I was the client what would I do about this?” Are the findings and recommendations useful? Discuss your initial findings and the implications informally with the client – your aim is to have no surprises when the answer is finally presented. Your insights are likely to stimulate client thinking. In turn clients may then broaden the scope of the project to include additional sources of information, additional analysis, or consideration of a broader range of options.

- Push beyond the first (and sometimes obvious) good answer. What are we missing? Where can we add greater value?

- Ensure that all the work being done by all the team members is brought together as a whole. You will probably be confident in your own work, but you also need to have confidence in the quality of work that has been done by team members.

- Before putting the final recommendations in front of the client you should test (at least a sample of the work done by others) for quality:
  - Are the conclusions based on logical analysis and accurate data?
  - What process was used to interpret the data?
  - What were the sources of the important pieces of data? Check for confirmation from a secondary source.
  - Have we used someone else’s conclusions or opinions as facts?
Synthesize findings and build argument

Work in Two Modes

Solve the problem

Data → Analysis → Synthesis → Message

Our task is not to do analysis, but to provide compelling, well-supported recommendations

Communicate the answer

Message → Synthesis → Analysis → Data
There are two main tools to synthesize and build your case:

Tool #1: for "one week answers"
- Situation
- Observation
- Implication

Tool #2: for "bullet proof case"

The Pyramid

Weekly cycle
1. Define problem
2. Build Issue Tree
3. Prioritise & Write Ghostpack
4. Develop Workplan
5. Gather Data & Analyse Critically
6. Synthesize & Iterate
7. Build Commitment

7-step approach to problem solving
Tool #1: Week one answers drive the project

**This**

- The problem or opportunity at the core of the decision-maker's dilemma
- The critical insight or leverage point is that is emerging
- The "logical highground"
- What set of actions is implied
- What options do we see at this point

**Not This**

- Company history or masses of facts unrelated to core opportunity / problem
- Vague description of complication (re-hash of problem statement) or another bunch of facts masquerading as insight
- "The" one and only answer
- Unsupported prejudices/ preconceived ideas not related to steps 1 and 2
Tool #2: Pyramid Structure to build bullet-proof case

- **Define problem**
- **Prioritise & Write Ghostpack**
- **Gather Data & Analyse Critically**
- **Develop Workplan**
- **Synthesize & Iterate**
- **Build Commitment**
- **Weekly cycle**

**Governing thought** – your answer to the audience’s question in a single statement

**Key line** – the core logic of your case

**Support** – on which your logic relies

*Use this when you have enough analysis and fact to begin to build a well supported case*
Influence the client

- Powerful communication of findings and recommendations, along with supporting arguments is vital to the success of a project. This communication is usually a powerpoint pack – started at the workplanning or data gathering stage.

- All packs are headed with a compelling story line and have a one to two page executive summary up front which explains “what the pack says”.

- Other charts that might be useful to present supporting evidence are set out on subsequent slides.

- Allow lots of lead time to prepare packs, enlist the help of management services staff who may have access to previous packs and templates.

- Socialise key messages with broad range of stakeholders and start gaining their commitment towards the recommendations/changes.

“The picture’s pretty bleak, gentlemen. … The world’s climates are changing, the mammals are taking over, and we all have a brain about the size of a walnut.”

Weekly cycle

1. Define problem
2. Build Issue Tree
3. Prioritise & Write Ghostpack
4. Develop Workplan
5. Gather Data & Analyse Critically
6. Synthesize & Iterate
7. Build Commitment
Why does this simple, common sense process go wrong?

“Oh, you wanted me to get data!”
- Yes, this does happen! Consultants come back with lots of war stories but NO DATA

“There is no data!”
- That sometimes means that the consultant does not know what data looks like
- More often it means the consultant is too complacent to roll his / her sleeves up and plough through reams and reams of paper to get at the data

Stating the obvious
- “Increasing margins will improve profits”
- We are looking for the ‘So-whats’ - keep on digging until you find it

Validate, validate, validate - "Oh, does that mean I should have validated my analysis?”
- Consultants are validation averse hypothesis-driven doesn’t mean fact-light
Good problem solving is very powerful!

“Wait a minute! Say that again, Doris! . . . You know, the part about, ‘if only we had some means of climbing down’.”