



“CURIOSITY NEVER KILLED ANYONE”


LANDSCAPING - THE SCIENCE OF QUESTIONING

SofTec Asia 2011, Bangalore

T Ashok

Founder & CEO, STAG Software

 in.linkedin.com/in/AshokSTAG

 [ash_thiru](https://twitter.com/ash_thiru)





CURIOSITY

We're all born with it.

VERY DEMOTIVATIONAL .com

Don't try this.
You could get killed!



The PROBLEM

How you understand expectations and needs?

If I do not have a deep domain knowledge, how do I understand an application to test well?

How do I come with good questions to understand?
Is there a scientific/systematic way to questioning?

Key premise

Good questioning is key to good understanding.

It is perfectly OK to ask questions for which no clear answers are available.

“It is more important to know what you do not know, rather than be content with what you know”

It is therefore important to come up with good questions rather than worry about the availability of answers.

The BIG QUESTION

Given an application and whatever available documentation (which is always less!) how do I understand well enough to test effectively?

Does my ability to understand depend solely on my prior experience with similar applications/domain?

What if I am not as well versed in that domain?



The story of Joe the tester..

In the story “The diagnosis”, Joe faced with the problem of understanding a new application in a domain that he is unfamiliar with, has the “Aha” moment at a doctors’s office during the process of ‘diagnosis’.

He sees parallels in doctor’s questioning technique to diagnose the problem and his problem of understanding of the application. He understands that decomposing the problem into information elements and establishing the connections between these elements enables him to come up with good questions to understand the application. Voila!

[Click here](#) for the story “The diagnosis”

Published in “Tea time with testers” May 2011

Joe figured that understanding an application is not just about walking through the various features via the user interface (assuming that the application has an UI).

It requires a scientific/systematic walkthrough of various elements commencing from the customer's needs/expectations and then into the application's deployment environment, architecture, features, behaviour and structure.

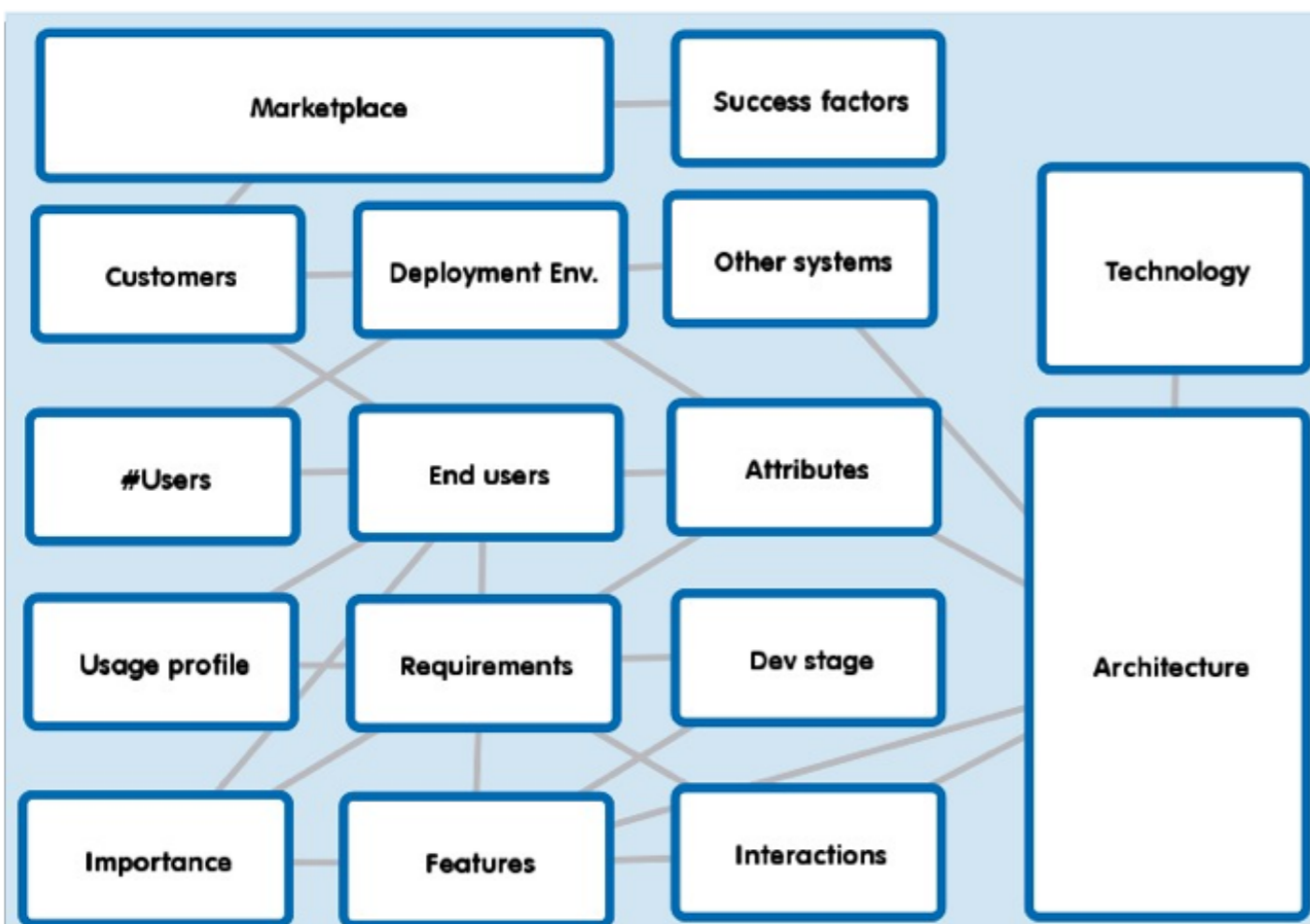
He understood that he needed to construct a "good landscape" of the system and establish a clear baseline for effective testing.

Landscaping - A core concept in HBT

Landscaping, a core concept in HBT (Hypothesis Based Testing) enables one to systematically come with meaningful questions to understand the end users, application and the context.

It is based on the simple principle:

“Good questions matter more than the answers. Even if questions do not yield answers, it is ok, as it is even more important to know what you do not know.”



Landscaping states that there are about SIXTEEN information elements that will enable good understanding of the system.

The act of seeking information of these SIXTEEN elements and their interconnections results in questions that aid in understanding.

Information needed for good understanding...

Success factors

The reason for deploying the system

Marketplace & Customer types

The target for our business

End user

Who will use our system? How many?

Requirement, Features, Attributes

What do they need? What are their expectations?

Ranking of features & Usage profile

Which is more important? How will it be used?

Interactions

How do feature(s)/requirement(s) affect each other?

Deployment environment

Where will it run?

Structure - Architecture, Technologies

How do the internals look like?

Stage of development

Built new or modified or status quo?

Behavior

What conditions govern the behavior of a feature?



Information needed for good understanding...

Success factors

Marketplace & Customer types

End user

Requirement, Features, Attributes

Ranking of features & Usage profile

Interactions

Deployment environment

Structure - Architecture, Technologies

Stage of development

Behavior

External information



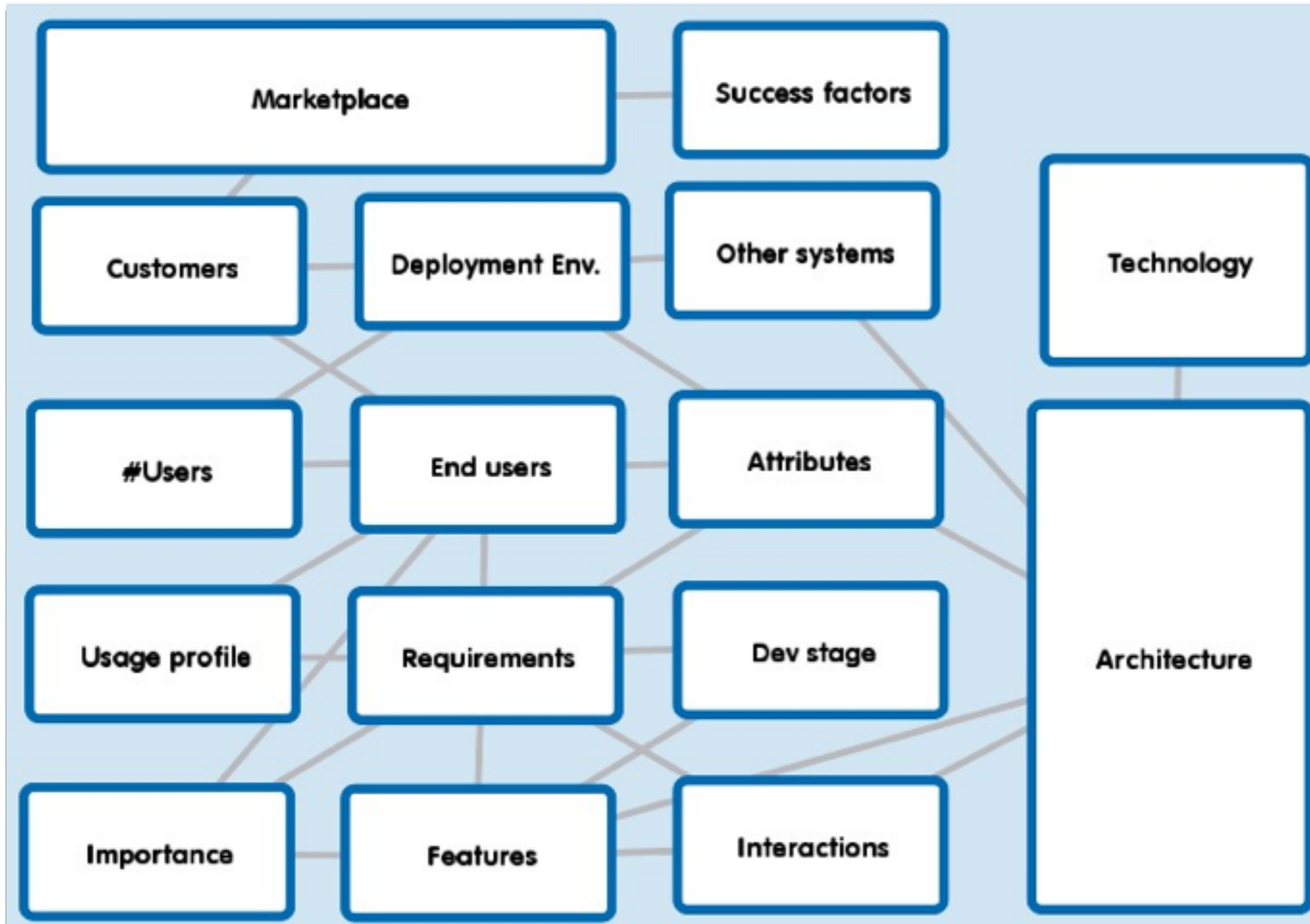
to

Internal information



Landscaping – A Core Concept in HBT

A technique to rapidly understand the system by examining the various elements and the connections between them.



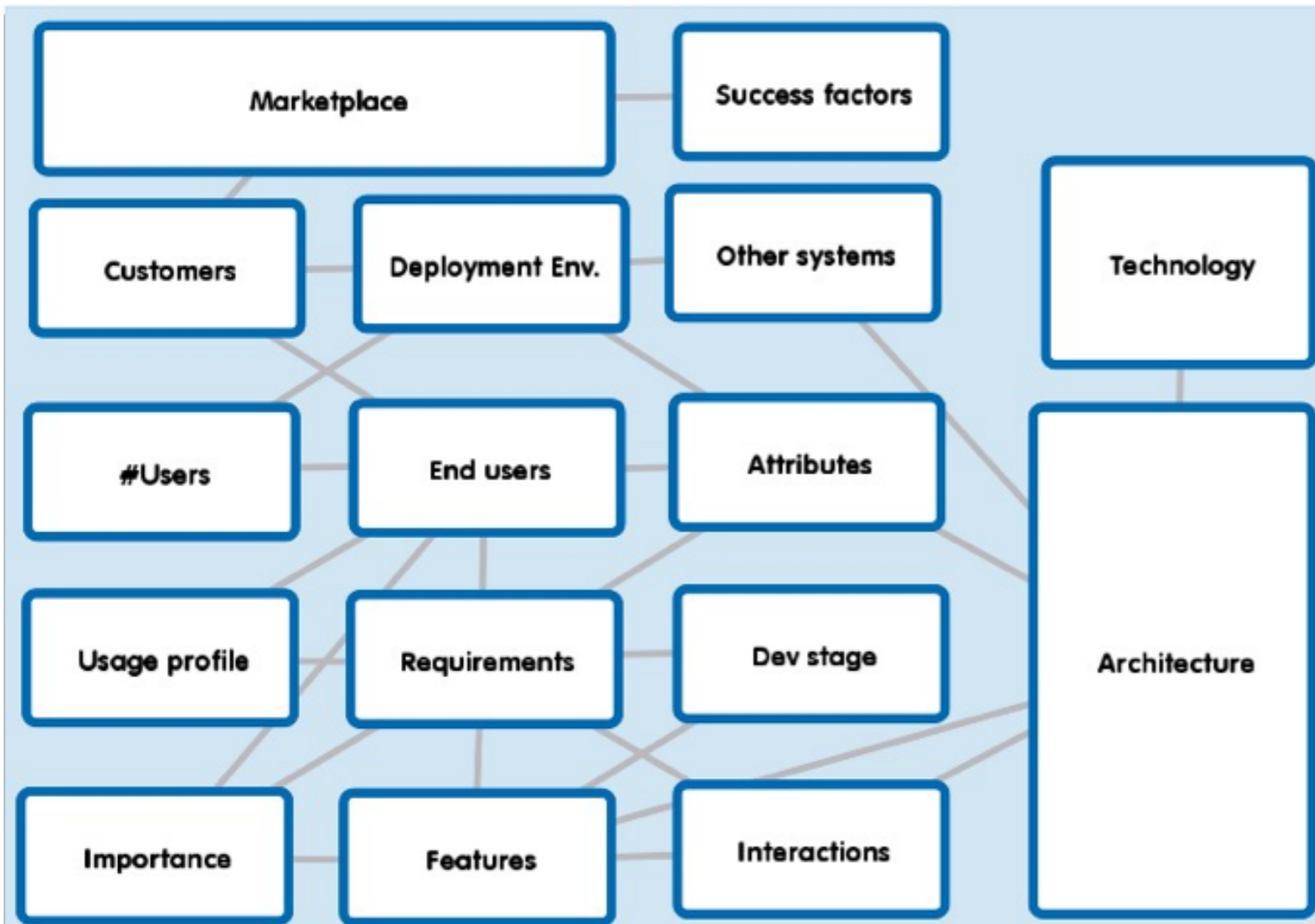
Some questions generated by applying Landscaping...

Marketplace	What marketplace is my system addressing? Why am I building this application? What problem is attempting to solve? What are the success factors?
Customer type	Are there different categories of customers in each marketplace? How do I classify them? How are their needs different/unique?
End user (Actor)	Who are the various types of end users (actors) in each type of customer? What is the typical/max. number of end-users for each type? Note: An end user is not necessarily a physical end user, a better word is 'actor'
Requirement (Use case)	What does each end user want? What are the business use cases for each type of end user? How important is this to an end user - what is the ranking of a requirement/feature?
Attributes	What attributes are key for a feature/requirement to be successful (for an end user of each type of customer)? How can I quantify the attribute i.e. make it testable?
Feature	What are the (technical) features that make up a requirement (use-case)? What is the ranking of these? What attributes are key for a successful feature implementation? How may a feature/requirement affect other feature(s)/requirement(s)?

Some questions generated by applying Landscaping...

Deployment environment	<p>What does the deployment environment/architecture look like?</p> <p>What are the various HW/SW that make up the environment?</p> <p>Is my application co-located with other applications?</p> <p>What other softwares does my application connect/inter-operate with?</p> <p>What information do I have to migrate from existing system(s)? Volume, Types etc.</p>
Technology	<p>What technologies may/are used in my applications?</p> <p>Languages, components, services...</p>
Architecture	<p>How does the application structure look like?</p> <p>What is the application architecture?</p>
Usage profile	<p>Who uses what?</p> <p>How many times does a end user use per unit time? i.e. #/time</p> <p>At what rate do they use a feature/requirement?</p> <p>Are there different modes of usage (end of day, end of month) and what is the profile of usage in each of these modes?</p> <p>What is the volume of data that the application should support?</p>
Behavior conditions	<p>What are the conditions that govern the behavior of each requirement/feature?</p> <p>How is each condition met - what data (& value)drives each condition?</p>

Summarizing...



Landscaping is a core concept in HBT (Hypothesis Based Testing) that identifies the various information elements, and the process of understanding the details of each element and their connections enables questions to arise.

These questions when answered allow one to understand the application(system), customer and the context, and establish a clear baseline for subsequent stages of testing.



We have also seen that when applying this, we uncover missing parts of the puzzle, this has helped us to improve/fix the requirements.

Good questions aid in early defect detection/prevention, we have used this to test requirements and not only code.



THANK YOU. HAVE A GREAT CONFERENCE.

Follow us  @stagsoft

Landscaping is covered in the HBT Series workshop “Rapid understanding of customer expectations” offered by CleanSoft Academy (A division of STAG Software) www.cleansoft.in

