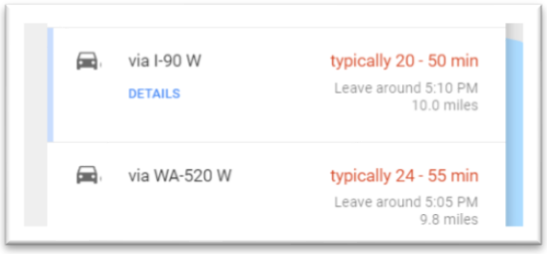
“When to Leave Home” Forecasting Exercise



Given two addresses compute the “**when to leave home**” time in order to arrive by a desired **target time**.

Q1 – What is the general algorithm you might use to compute this time?

Spend 10 minutes discussing the problem and how you will compute the “leave home” time. Then decide on one person to present that algorithm using step-by-step sticky notes to the room in 2 minutes or less. Tip: start by discussing how you personally decide when to leave home when travelling to a fixed time event.

Extra credit: Define “success” for your solution.

After hearing and discussing all of the group’s ideas, spend 5 minutes and decide what method you think solves this problem the best.

Q2 – What data would you require to implement that algorithm?

Spend 5-10 minutes creating a sticky-notes for each input data or information required for your algorithm. Decide if its critical to have (can’t do without) or if it is an optimization (improves the forecast) and then decide on one person to present those stick-notes to the room in 2 minutes or less.

Extra Credit: For the critical data, describe what you will do if you can’t get or don’t yet have that data.

Q3 – What factors could cause forecasting error (arriving late)?

Spend 5-10 minutes discussing what external factors could cause forecasting error. Create a sticky-note for each factor and describe how you might “deal” with that factor in your forecasts. Decide on one person to present those sticky-notes to the room in 2 minutes or less.

Q4 – How might you measure success of this feature in production?

Spend 5 minutes discussing how to measure success in production, and how production data could improve the results over time. Decide on one person to present your findings to the room in 2 minutes or less.

Q5 – How might this same algorithm be used in forecasting when you need to start software features to deliver by a given date?

Spend 15 minutes as a group discussing how this technique might be used for forecasting software projects or features in multiple team environments. For example; What would need to be changed? What data you need for the feature/project and the system delivering those features/projects?

Feedback: troy.magennis@focusedobjective.com

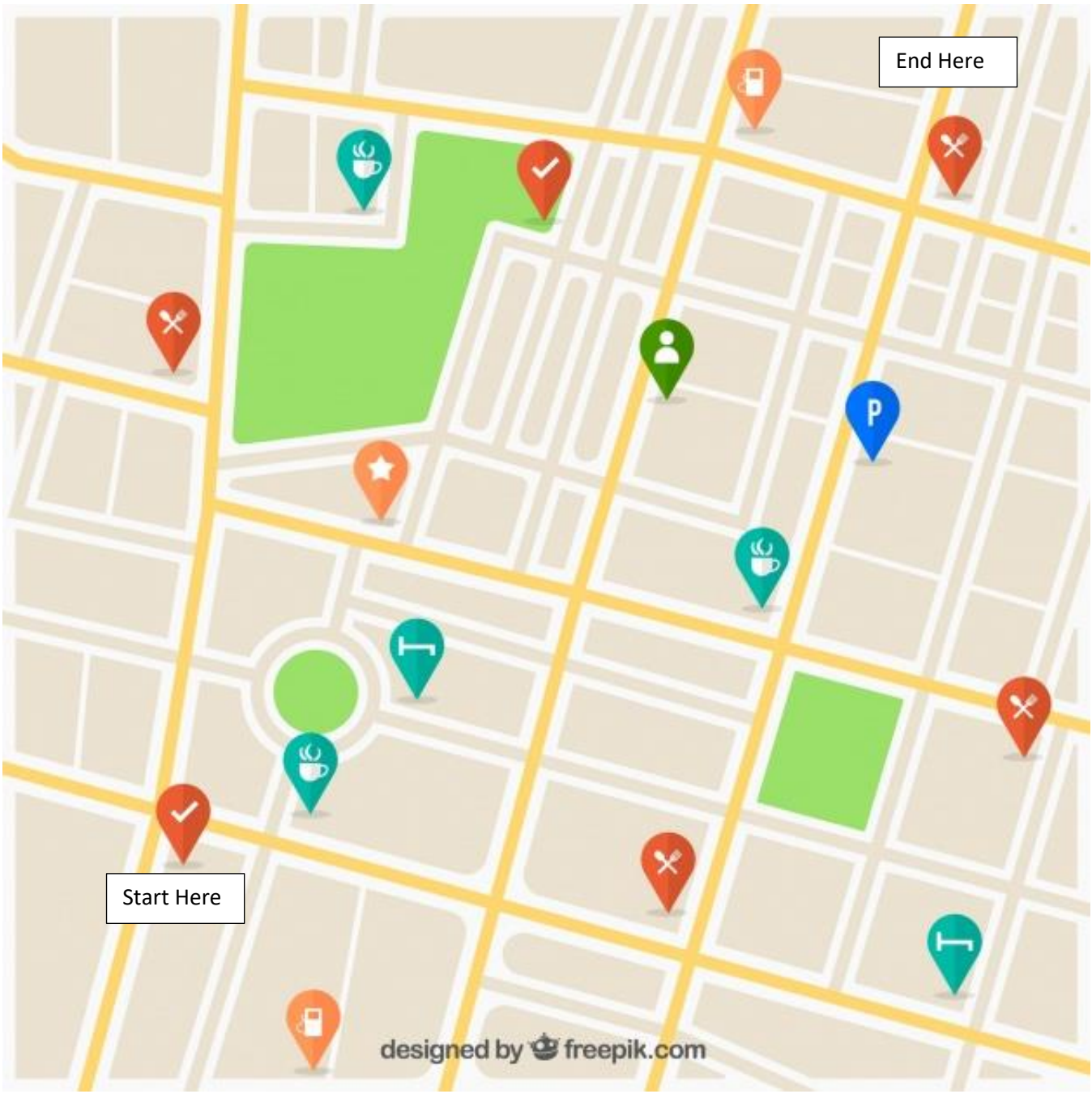


Figure 1 - Example road network diagram to use for discussing your algorithm.

Image attribution: Travel vector created by freepik - www.freepik.com

Facilitation guide

Open the exercise by describing the goal, forecasting when to LEAVE to arrive by the Target Time.

Facilitate a quick discussion about Google Maps or GPS navigation as solutions. If the group is less technical, discuss how a network of road data might be used as the basic input data. Just facilitate enough to kickstart the algorithm group discussion. Avoid solving it for them (if you can help it!)

Q1. Algorithm

Give the groups 10 minutes to discuss the broad high-level algorithm that might get a result. If groups struggle to start, get them to write down the manual process they would use to compute time to leave.

Debrief – get each group to explain their approach using stick-notes as a sequence. Align similar steps vertically or horizontally.

Example:

1. Get list of "all" ways from point A to B
2. Score each option (ask how they would score each option)
    a. Listen for # lanes, speed limit, # lights, # stop signs, historical travel times, etc.
    b. Listen for when they say use data, ask what they would use if no data yet
3. Select the best (ask what "best" means to this group)
    a. Listen for how they assess best.
4. Subtract travel time from desired target arrival time
    a. Listen for how they incorporate uncertainty, e.g. Average or percentile

Give each group a few minutes to capture changes to their approach.

Learning objective: Many options with conflicting benefits. You need to score to choose.

Q2. Data

Give the groups 5-10 minutes to list data required. Use a different color sticky-note for necessary and supplemental data.

Debrief – get each group to show the data they need. Map the groups data near the algorithm steps from Q1.

- Challenge for ways to estimate that data until you have real world data.
    - Speed limit. But do cars travel at the speed limit? Contrast full team capacity?
- Challenge for data quality issues, e.g. Seasonality, time of day, special events, etc.
    - Recent in context data is superior to just any data. Don't use peak hour data for weekend travel time predictions!

Learning objective: There is data to get any result and data that improves result. Learn about the context of data and how some data needs to be excluded. Important to be able to have a safe default if no data.

Q3. Impacting Factors

Ask the groups to spend 5-10 minutes brainstorming and writing on sticky-notes reasons the forecast might turn out wrong. Ask them to stack rank those factors based on impact. Walk around, if groups aren't detailed with factors, e.g. "slow travel" press them for what causes slow travel.

Debrief – get each group to show the factors, and press them for their ideas on minimizing the chance or impact of these factors. How can they tell a route is likely to encounter this versus not?

Factors expected: Peak hour, direction of travel, accidents, weather, sporting events, bridges, tolls, traffic signals, train intersections, commuter lanes.

Learning objective: These can have huge impact. How do we anticipate? How do we incorporate?

Q4. Measuring Success (and Improving)

Give the groups 5 minutes to discuss what data they could use from production to affirm and improve their algorithm and use of data.

Debrief – get each group to present their criteria. The easy answer is you arrive before the target time. Prompt for arriving early is also bad (if you like family or sleep time). Discuss how impact of being late changes the risk tolerance and how you might determine how tight is too tight. E.g. if the target location is an airport or school during the afternoon, then maybe leave more time than a restaurant or shopping mall.

Debrief Challenge for Entire Group: If you didn't have a target date, what would you measure success against? Smart groups might say, against average travel time, but the key point is without a target date, success is wishy washy.

Learning objective: Open ended forecasts are difficult to learn from, as no success feedback.

Q5. Pivoting to Software Forecasting

The goal of facilitating this as a group is to pivot thinking into how we forecast software features and projects.

Key points you are listening for, and might need to prompt for –

1. Have a target date before starting otherwise its hard to gauge success and use that data to as feedback to improve.
2. There are multiple route options, so we should consider multiple delivery options.
3. Scoring options is necessary to know why you chose the one you did – when do you commit?
4. Defining success is important, too early is sometimes as bad as too late – when?
5. Different forecasts have different level of impact if "wrong" – how do you decide?
6. Sometimes - Impacting factors have greater impact than travel distance – when?
7. Arrival time updates when started in Google maps, how might you do that with your forecasts?
    a. Is Google maps forecast wrong based on an unforeseen accident?
    b. Is Google maps forecast wrong for not catering for peak-hour travel times?
8. Handoff to different teams may be impacted by "traffic" depending on time of use (just like peak hour traffic)