Software-Defined Policies: An Introduction

By Bala Balabaskaran

Founder and CTO, fullcast.io

fullcast.io

Transforming Sales Operations into Growth Operations

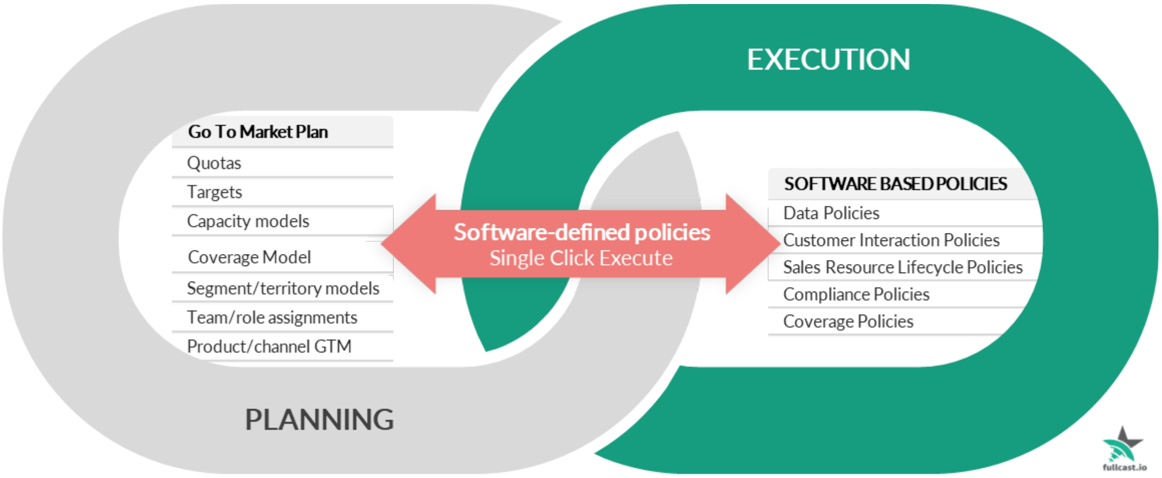
Table of Contents

Software-Defined Ops

Software-defined ops is the idea that all aspects of operation can be programmatically defined and executed as a set of policies through application programming interfaces (API) and translate strategy into execution.

Today, businesses struggle to translate strategic intent into standard operating procedures. This lack of agility and disconnect between strategy and execution is typically handled manually. In a fast-moving business environment, the ability to react to business conditions and respond operationally is the difference between well run and successful organizations and those that are not.

This approach takes a page from a variety of software-defined initiatives in the DevOps movement which has successfully translated cloud infrastructure into software-defined networks and software-defined data centers.



What is a Policy?

Wikipedia Definition:

A policy is a deliberate system of principles to guide decisions and achieve rational outcomes. A policy is a ***statement of intent*** *and is* ***implemented as a procedure or protocol****. Policies are generally adopted by a governance body within an organization. Policies assist in both subjective and objective decision making. Policies in subjective decision making assist senior management with decisions that must be based on the relative merits of a number of factors, and as a result, are often hard to test objectively, e.g. work-life balance policy. In contrast, policies in objective decision making are operational in nature and can be objectively tested, e.g. password policy.*

Let’s break that down:

1) It is a statement of intent which describes a set of principles in the form of a procedure or protocol. Typically, the principles are defined by your business strategy and are translated into a series of operational procedures that are then executed by various operational teams across the business.
2) Policies need to be adopted by a governance body within the organization. These are typically the various line of business operational teams whose roles are to evaluate whether these procedures and protocols are followed and continuously report on their effectiveness.
3) Some policies require human intervention because there is subjectivity involved in the decision making around the evaluation and application of a policy. Objective policies can automate the evaluation and application.

Policies function in the software-defined ops world in 3 ways:

1) Policy Setting – Where the intent is defined
2) Policy Control – Where the policy is evaluated against the going concern (i.e. the regular operational motion of the business)
3) Policy Action – Where a required procedure or protocol is activated based on the evaluation

So what does thinking about this as policy instead of a set of ad-hoc processes get you? Let’s borrow from Wikipedia again:

1. *Avoids risks (financial losses from incomplete and uncoordinated exploitation of information, wasted time, failures of innovation, and reputation loss)*

2. *Achieves positive benefits, including negotiation and openness among those responsible for different aspects of information management*

3. *Productive use of IT infrastructure*

4. *Ability to initiate change to take advantage of changing environments*

5. Repeatability so as to achieve consistent outcomes every time

Stages of Policy Setting

Virtually every business policy goes through five main development stages. Those stages are 1) problem identification, 2) formulation, 3) decision-making, 4) implementation, and 5) evaluation. Each stage provides a measure of guidance that works together to improve productivity and strengthen the bottom line. James E. Anderson provides a great explanation of this process in *Public Policy-Making* (1974) where he details the policy stages:

- Agenda setting (problem identification) - The recognition of certain subject as a problem demanding further attention
- Policy formulation - Explores a variety of options or alternative courses of action available for addressing the problem (appraisal, dialogue, formulation, and consolidation)

- Decision-making - Decides on an ultimate course of action, whether to perpetuate the policy status quo or alter it (decision could be 'positive,' 'negative,' or 'no-action')
- Implementation - The ultimate decision will be put into practice
- Evaluation - Assesses the effectiveness of a policy in terms of its perceived intentions and results; Policy actors attempt to determine whether the course of action is a success or failure by examining its impact and outcomes

Simple enough. How often this cycle works is based on the type of operational process. For sales, which is a daily business and constantly evolving and changing, especially in a high-growth environment, the responsiveness of this cycle is very important to a high-functioning sales organization.

Types of Operational Policies Relevant to Sales

If we look at the world of sales operations, there are a few types of policies relevant to sales required to keep the sales organization humming along.

Data Policies – CRM Data

Data policies are meant to govern the quality of the data in the sales process, specifically your CRM. Data policies are meant to achieve the following goals as it relates to the data:

Accuracy

- Data should be sufficiently accurate for its intended purposes
- Data should be captured on the principle of 'getting it right the first time,' so data is captured once, as close to the point of activity as possible, with clear and simple actions and limited manual intervention (eg, administration, data cleansing)
- Accuracy is likely to be higher if staff providing data are aware of its importance and quickly have access to information, especially if there is a benefit for securing the quality of that data, such as receiving relevant performance information in return
- Accuracy is a balance between the importance of data and the cost of collection; Where there are compromises on accuracy for the sake of cost, the limitations of data should be clear to the users; Compromise is unlikely to be appropriate in the case of data supporting published performance indicators

Validity

Data should be recorded and used consistently, in compliance with relevant requirements, rules, or definitions.

Reliability

Data should reflect stable and consistent data collection processes, or where changes or differences are necessary, sufficiently documented and understood so information produced is stable and consistent.

Timeliness

Data should be captured as quickly as possible and must be available for its intended use within a reasonable time period.

Completeness

Identification and highlighting missing, incomplete, or invalid records can provide an indication of data quality and highlight ways the data capture process can be improved.

Customer Interaction Lifecycle Policies

The customer interacts with your organization in a series of interactions through sales, service, and customer support. These interactions have a set of policies that govern their lifecycle and the data collected through this process. These interactions are commonly referred to as the Campaign to Renewal Process. This is not a linear process and consists of a set of smaller interactions that are captured in the CRM as follows:

- Campaigns
- Leads
- Opportunities
- Quotes
- Contracts
- Orders
- Subscription (or the delivery engagement)
- Renewal

Throughout this process, there are a set of policies that govern:

- The stages and entry and exit criteria for each stage; The best example of this is opportunity stages and conditions met for opportunity creation and completion
- The valid state transitions and lifecycle events that trigger these transitions; For example, how does non-payment affect service delivery? What grace periods are offered? Who needs to approve discounts above a certain amount in a quote being sent to a customer?
- The contact roles engaged at each of these interactions and the respective stages
- Minimum data capture requirements at each stage and state transition and any cross-process validations; For example, you should not be able to move an opportunity to Closed Won stage if the contract has not been signed
- Interactions that miss the above processes will need to be handled in an appropriate manner such as a Closed Lost opportunity; We migrate the data from opportunity to the customer record so appropriate nurture activities can be executed

Sales Resources Lifecycle Policies

Lifecycle events can be related to salespeople, roles, people in roles, business going out of business, contacts changing roles, etc. These lifecycle events can impact how the lead to renewal process responds.

These Lifecycle events impact the lead to renewal policies. For instance, what happens to open opportunities when the rep working it leaves the business? What do you do with accounts no longer covered? Who gets credit for deals on uncovered accounts? When can new roles be introduced?

Compliance Policies

Compliance policies deal with adherence to regulatory and legal requirements. It is possible for various countries, states, and counties to have regulations for doing business in their jurisdictions. Examples of these policies include:

- Data Privacy Regulations – Such as GDPR in the EU or Privacy Shield from the US Dept of Commerce; Regulations require that the use of contact data be governed against misuse
- Trade Regulations – Governments may identify organizations, countries, and individuals you cannot do business with
- Accounting Principles – Various accounting principles such as ASC606 govern the collection of appropriate data for accounting, revenue recognition, or commissioning purposes; These policies must be followed in the course of doing business and collecting data
- Auditing Requirements – Your internal auditing requirements will require you to follow certain policies to ensure proper compliance with internal audit controls
- Industry Regulations – There are various industry regulations

Coverage Policies

The go-to-market model and the coverage model will dictate roles and interactions with customers and which CRM detail is delivered to each role. Coverage policies determine what information is delivered to which roles for job performance. This covers everything from round-robin delivery of leads to your sales development teams to the assignment of accounts to sales reps as part of their defined territories, etc.

Policy Setting and Go-To-Market Planning

The various policies mentioned above are typically a direct result of the go-to-market strategy. The intent around how the operating system should function is a direct result of the agenda set by the go-to-market strategy. Therefore, while planning your go-to-market, it's important to express the intent of the operational procedure or protocol.

When the go-to-market agenda is changed, it's important the corresponding operational policies are changed immediately. This is where the software-defined idea comes into play. Now you don't need to have data loads or wait for the IT department's backlog, etc.

A software-defined system will keep the two in sync.

Policy Evaluation

The policy evaluation is done using an agent that performs this operation within a transactional system. In the case of a CRM like Salesforce, this would take the form of a managed app. In other cases, these agents can be built external to the various transactional systems where the day-to-day operational concern is tracked. Go-to-market policies in a software-defined ops environment can be translated by these agents into CRM systems, financial systems, HRIS systems, commissions systems, Order Management Systems, Partner Management Systems, CPQ systems, etc.

Policy Action

Once the policy is evaluated, there could be one of two actions taken:

1) Alert – Alert the respective actors within the operational teams of failed or successful policy evaluation
2) Take an Action – This can take one of two forms:
    a. Complete automation action – e.g. assignment policies based on territory design or holdout stamping on opportunities on an account owner change
    b. Human involved action – e.g. discounting process requiring approvals

It is important to make sure there is an audit trail of the result of the policy evaluation, the appropriate action taken, and the parties involved. This will allow for the next step in the process, which is to evaluate whether the policy is having the desired effect.

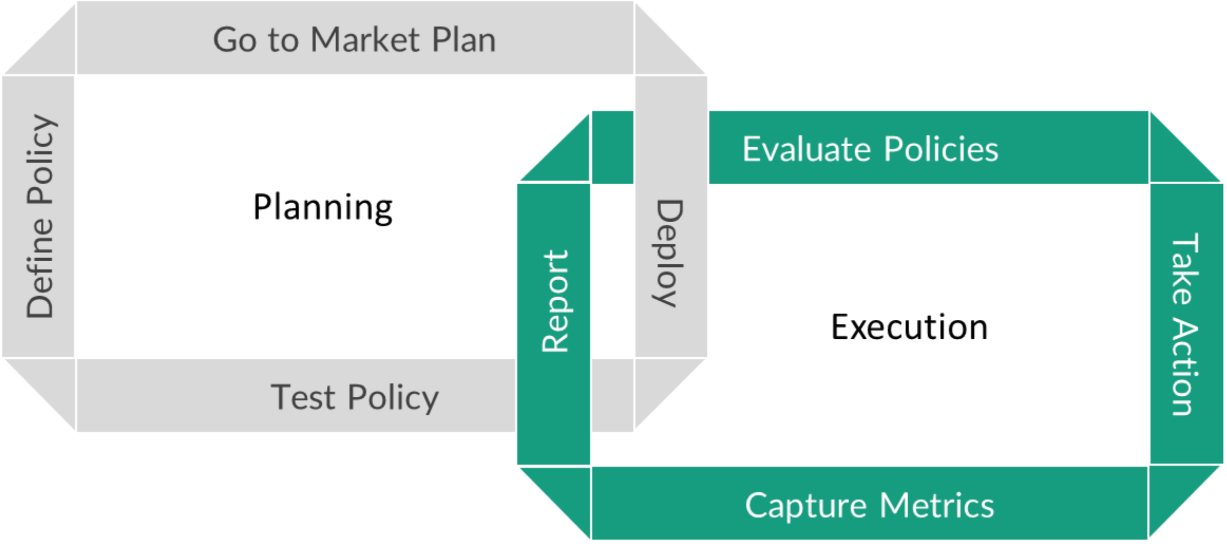
Policy Reporting & Auditing

It's important to keep a very close eye on the success of the policy intent and whether it's achieving the desired results for the organization. It's important to understand the key success metrics that each intended policy is looking to improve.

Reporting is a critical component of ensuring the continuous process of improvement is enabled.

Architecture for a Software-Defined Ops Platform

The architecture of a software-defined ops platform should support the definition of the policy in a way that it can be extended to cover the various aspects of policy definition, evaluation, and automated actions taken.

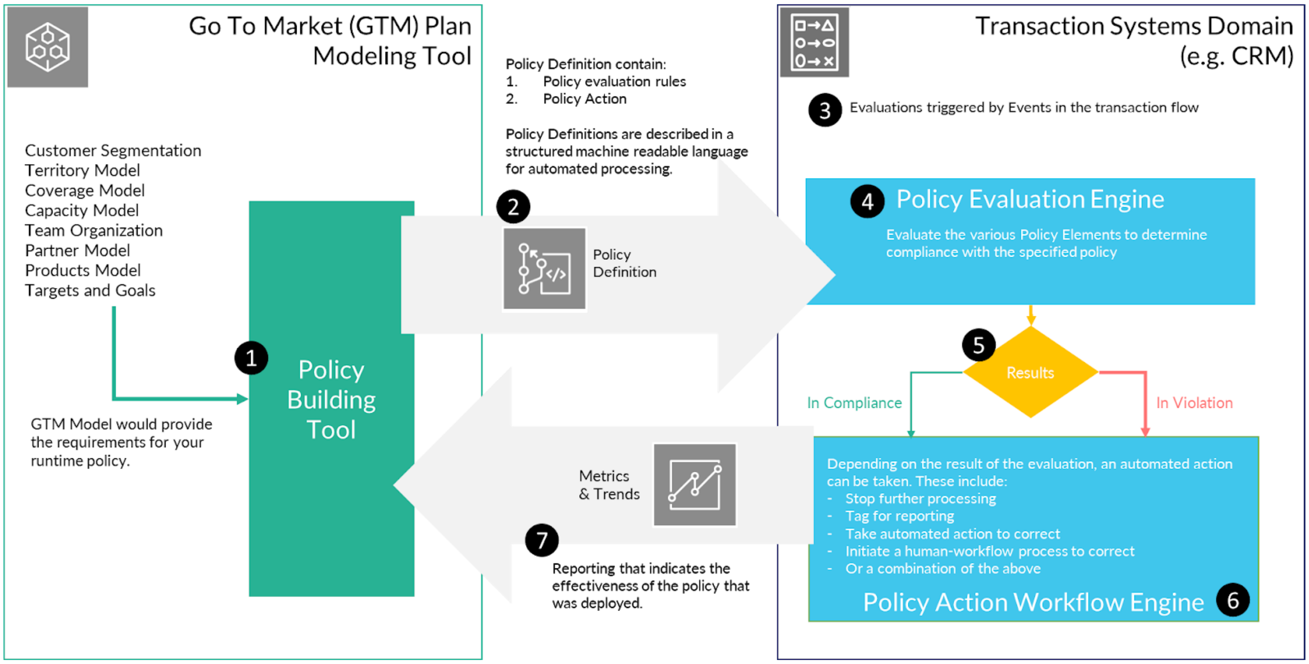


To automate this process, the following elements are critical in a software-defined ops framework:

1) Policy Definition Language – A way to define policies in an extensible manner that is machine readable
2) Policy Building Tool – Ideally integrated into the go-to-market planning process and tools; This tool should help you automatically specify the policy associated with the various aspects of your go-to-market plan

3) A Policy Evaluation Engine – Triggered based on the appropriate event in your day-to-day execution to evaluate the policy defined; The evaluation should indicate whether the policy conditions were satisfied or violated
4) An Action Engine – Take the necessary steps to either remedy the issue or incorporate a human workflow process to address it; The action could simply be reporting a violation of a policy to a complex set of operations such as taking corrective actions or preventing the process from going any further

Policies should be enabled on an as-needed basis and as the maturity of the organization and the complexity of the sales process demands it.


Go To Market (GTM) Plan Modeling Tool
Customer Segmentation
Territory Model
Coverage Model
Capacity Model
Team Organization
Partner Model
Products Model
Targets and Goals
GTM Model would provide the requirements for your runtime policy.
1
Policy Building Tool
Policy Definition contain:
1. Policy evaluation rules
2. Policy Action
Policy Definitions are described in a structured machine readable language for automated processing.
2
Policy Definition
Transaction Systems Domain (e.g. CRM)
3
Evaluations triggered by Events in the transaction flow
4
Policy Evaluation Engine
Evaluate the various Policy Elements to determine compliance with the specified policy
5
Results
In Compliance
In Violation
Depending on the result of the evaluation, an automated action can be taken. These include:
- Stop further processing
- Tag for reporting
- Take automated action to correct
- Initiate a human-workflow process to correct
- Or a combination of the above
Policy Action Workflow Engine
6
Metrics & Trends
7
Reporting that indicates the effectiveness of the policy that was deployed.

(1) The policy building is typically associated with the definition of the go-to-market plan for the company. This process provides the requirements for the policy to be followed during the runtime. So during the planning and modeling process, the Policy Building Tool can create a machine-readable policy.
(2) This policy is expressed using policy definition language that captures (a) the rules to evaluate if a particular aspect is in compliance or in violation of a policy, and (b) the automated action based on the result of the evaluation.
(3) The policy is then transported through the Application Programming Interface (API) to the various transactional system(s). A transaction system is where your day-to-day go-to-market is captured, recorded, and managed. For most companies, this is a CRM system.
(4) Policy evaluations are triggered by various actions during the course of the sales and support motion as defined in the policy definition in (2) above.
(5) The evaluation of the policy either yields a result which indicates if the aspect evaluated is compliant. The policy definition in (2) above also describes the action to be taken.

(6) The Policy Action Engine then triggers the appropriate action to either correct or observe and report.
(7) The policy evaluation results and trends are then measured extensively and reported back to the policy definition process and tool so the effectiveness of the policy can be understood, and any necessary changes made.

Rinse and Repeat.

Conclusion

Applying a policy-defined lens on sales operations brings together the go-to-market design process and the execution process through an automated mechanism. Human intervention to interpret and implement the go-to-market into the transactional systems is no longer needed. The automated actions prevent the operations teams from getting caught in an endless cycle of data quality cleanups and manually fixing unalignment with your go-to-market.