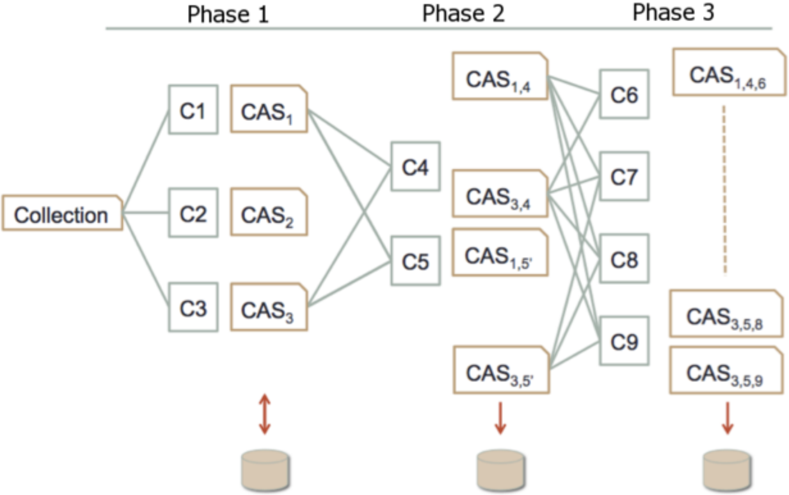
Phase 1
Phase 2
Phase 3
Collection
C1
C2
C3
CAS$_1$
CAS$_2$
CAS$_3$
C4
C5
CAS$_{1,4}$
CAS$_{3,4}$
CAS$_{1,5'}$
CAS$_{3,5'}$
C6
C7
C8
C9
CAS$_{1,4,6}$
CAS$_{3,5,8}$
CAS$_{3,5,9}$

Definition: Phase, Component, Parameter, Configuration I

- Most information systems consist of a number of processing units or components arranged in series, and each component is described by its input(s) and output(s).

Example

- A typical question answering system has four main component types: question analyzer, document retriever, passage extractor, answer generator [27].
- A typical ontology-based information extraction pipeline will integrate several preprocessors and aggregators [33].

Definition: Phase, Component, Parameter, Configuration II

- These processing steps can be abstracted as phases and stages in a pipeline.

Definition (Phase, component, parameter, configuration)

- The processing unit as the t-th step in a process can be conceptualized as a **phase** t.
- A **component** f_t^c in phase t is an instantiated processing unit, which is associated with a set of **parameters**, denoted by $\{\omega_t^{c,p}\}_p$, which constitute a component **configuration** ω_t^c.

Definition: Phase, Component, Parameter, Configuration III

Example

- In Question Named Entity Recognition, a phase t in a question answering system where the input x_{t-1} is a question sentence, and the output x_t is a list of named entities.
- Component f_t^1 could be a rule-based named entity extractor, f_t^2 could be a CRF-based named entity extractor, and f_t^3 could be a named entity extractor based on knowledge base lookup. Configuration parameter value
- ω_t^1 could be the set of rules to use, ω_t^2 could be a weight trained for the CRF model, and ω_t^3 could refer to the knowledge base to be used by the component.

Definition: Phase, Component, Parameter, Configuration IV

- Two important characteristics of the configured component.

Definition

- *Cost* of resource required to execute the component on input x: $c(f_t^c | \omega_t^c, x)$
- *Benefit* of executing the configured component to performance improvement: $b(f_t^c | \omega_t^c, x)$

- Resources used by a component include execution time, storage space, network bandwidth, etc., which can be measured by CPU time, allocated memory size, and data transfers respectively.

Definition: Phase, Component, Parameter, Configuration V

- A resource utilization measure can also be a more specific function of component characteristics (e.g., the cost to execute a configured component on Amazon Web Services[5] is a function of execution time and hardware capacity utilized).

[5] `http://aws.amazon.com/`

Definition: Trace and configuration space I

- A typical information processing task can be described as *n* processing phases arranged sequentially.

Definition (Trace and configuration space)

- A **trace** $\mathbf{f^c}|\omega^{\mathbf{c}}$ is an execution path that involves a single configured component for each phase, which is formally defined as $(f_1^{c_1}|\omega_1^{c_1}, f_2^{c_2}|\omega_n^{c_2}, \ldots, f_n^{c_n}|\omega_n^{c_n})$.
- The set of all components with all configurations comprise the **configuration space** $\mathscr{F}|\boldsymbol{\Omega} = \{\mathbf{f^c}|\omega^{\mathbf{c}}\}_c$, and a subset $F|\Omega \subseteq \mathscr{F}|\boldsymbol{\Omega}$ is referred to as a **configuration subspace**.

Definition: Trace and configuration space II

Example

- Question analyzers, document retrievers, passage extractors, and answer generators comprise the configuration space for a typical four-phase question answering task.
- One single execution path would be a unique combination of components (e.g. “Query tokenized by white-space string splitter, document retrieved from Indri repository index with default parameters, sentence extracted based on LingPipe sentence segmenter and VSM (Vector Space Model) similarity calculator”) or a trace in the configuration space.

- Extension of *cost* and *benefit* for a trace and a configuration subspace

Definition: Trace and configuration space III

Definition

- The cost to execute a trace is the sum of costs to execute each configured component.

$$c(\mathbf{f}^{\mathbf{c}}|\omega^{\mathbf{c}}, x_0) = \sum_{t=1}^{n} c(f_t^{c_t}|\omega_t^{c_t}, x(c_1, \ldots, c_{t-1})) \tag{1}$$

  where $x(c_1, \ldots, c_{t-1})$ represents the output from a series of executions (or a partial trace) $(f_1^{c_1}|\omega_1^{c_1}, \ldots, f_{t-1}^{c_{t-1}}|\omega_{t-1}^{c_{t-1}})$.

- The performance of a trace corresponds to the final output from last execution.

$$b(\mathbf{f}^{\mathbf{c}}|\omega^{\mathbf{c}}, x_0) = b(f_n^{c_n}|\omega_n^{c_n}, x(c_1, \ldots, c_{n-1})) \tag{2}$$

Definition: Trace and configuration space IV

Definition

- The cost of the entire configuration subspace is defined as the sum of unique executions of configured components on all outputs from previous phases.

$$c(F|\Omega, x_0) = \sum_{t=1}^{n} \sum_{c_1=1}^{m_1} \cdots \sum_{c_t=1}^{m_t} c(f_t^{c_t}|\omega_t^{c_t}, x(c_1, \dots, c_{t-1})) \tag{3}$$

- The benefit of the configuration space is defined as the benefit of the best-performing trace.

$$b(F|\Omega, x_0) = \max_{\mathbf{f}^{\mathbf{c}}|\omega^{\mathbf{c}} \in F|\Omega} b(\mathbf{f}^{\mathbf{c}}|\omega^{\mathbf{c}}, x_0) \tag{4}$$

Definition: Configuration space exploration I

Definition (Configuration space exploration)

- For a particular information processing task, defined by
  - m_t components for each of n phases: $f_t^1, f_t^2, \ldots, f_t^{m_t}$, with
  - corresponding configurations $\omega_t^1, \omega_t^2, \ldots, \omega_t^{m_t}$, given
  - a limited total resource capacity $\mathscr{C}$ and
  - input set $\mathscr{S}$,

  configuration space exploration (CSE) aims to find the trace $\mathbf{f}^{\mathbf{k}} | \boldsymbol{\omega}^{\mathbf{k}}$
  - within the configuration space $\mathscr{F} | \boldsymbol{\Omega}$
  - that achieves the highest expected performance without exceeding $\mathscr{C}$ of total cost.