

Human Friendly Computing: Neural-like, self-organizing data and functions *Communicating with the Machine*

Rocky Nevin, CEO/CTO DataSea Inc.

Before we communicate with a person
we must communicate with the Machine

... improve!

Why do we have to do the work?

We must guide it to the next step

Von Neumann, c. 1945:

- *Paraphrased: Do things in steps, and tell me the address, and I'll tell you what's there.*

Why do we have to do the work?

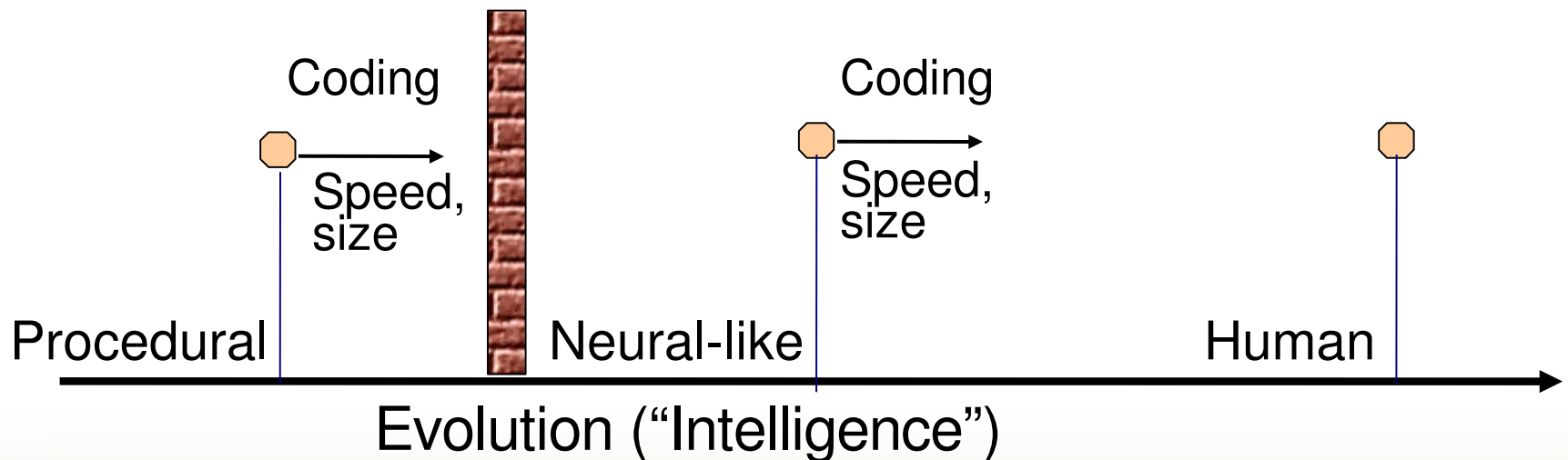
Machines are simple: evolved from tabulators

Tabular data structures

isolated data

Tabulating algorithms

```
mov ax, 09
```



- * Humans ≠ Computers
- * Language has evolved for a reason

The logo for eComm 2008, featuring the word "eComm" in a stylized font with a swoosh over the "e", followed by "2008" in a bold, sans-serif font.

eComm™2008

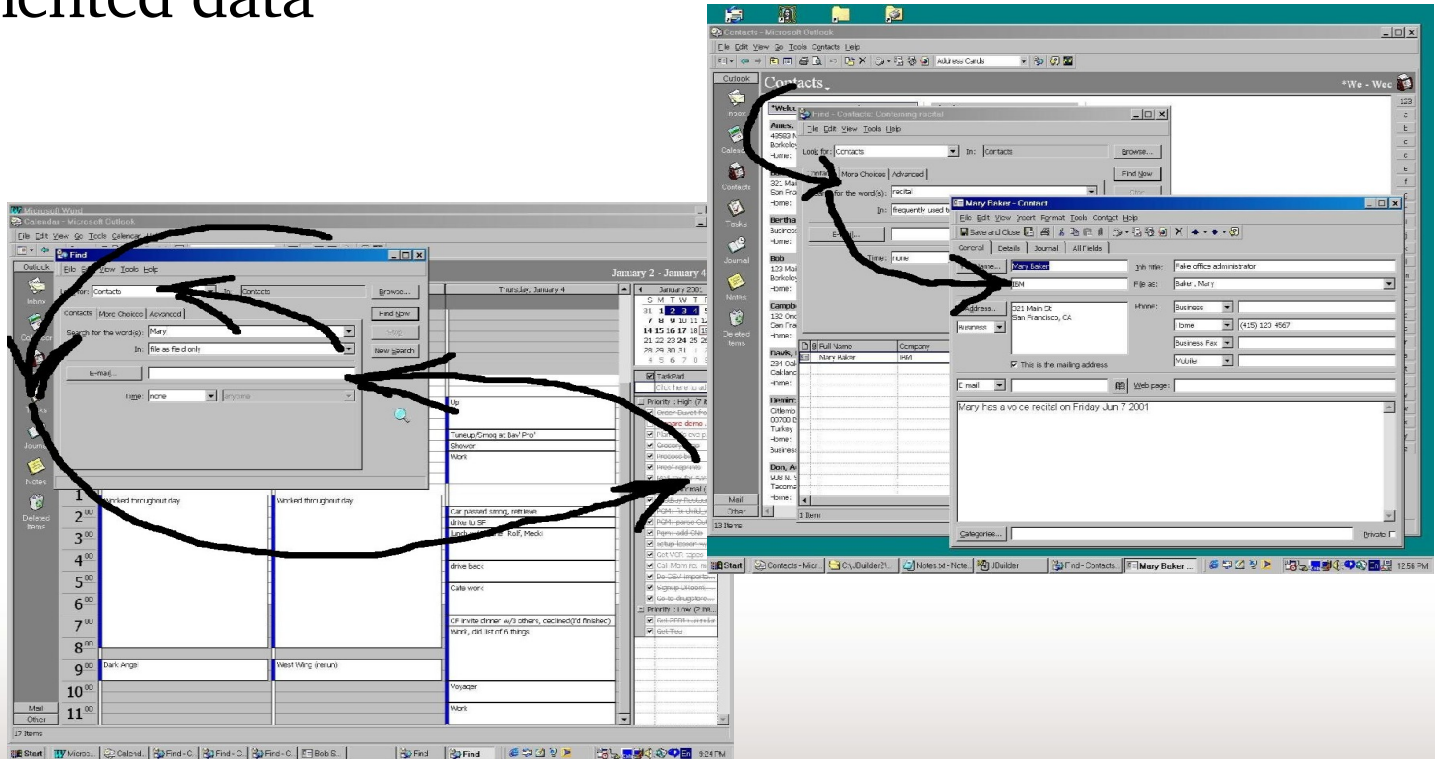
Emerging Communications Conference

THE TRILLION DOLLAR INDUSTRY RETHINK™

“I know what I want,
why can't I just ask for it?”

Today's Look: Clutter

- Pulldowns, decision-trees
- Fragmented data



Today's Look: Hieroglyphs

Old Model

- Late 1st Pharaonic Dynasty Hieroglyphs?

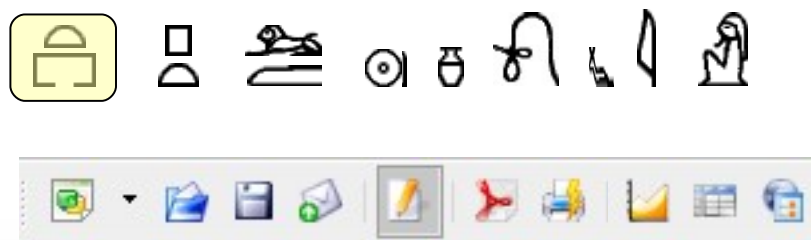
Which one represents “home”?



Today's Look: Hieroglyphs

Old Model

- Late 1st Pharaonic Dynasty Hieroglyphs?

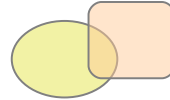


Four models of interaction

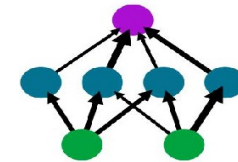
1) Follow Trees: pre-defined pull-downs (most app's)



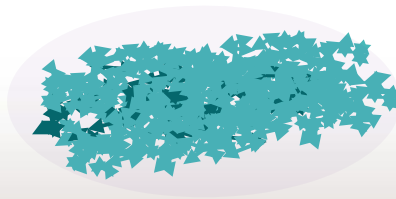
2) Intersection of sets: e.g. search engines



3) Neural Networks: optimization, processing layers



4) 'Neural-like' processing: DataSea



Four models of interaction: Trees

1) Follow Trees: pre-defined, pull-downs (most app's)



we learn and adapt

Forced decisions

many steps,

islands of data

====> "Answer not found", "Error",

"Operation not allowed"

2) Type our queries (e.g. search engines)

3) Neural Networks: optimization, processing layers

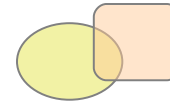
4) 'Neural like' processing: DataSea

Four models of interaction: Intersections

1) Follow Trees: pre-defined pull-downs (most app's)

2) Type our queries (e.g. search engines)

Keywords, some Natural Language
Queries only **commands & new info**
Answers are 'hits' ... **answers** ...
not enough!



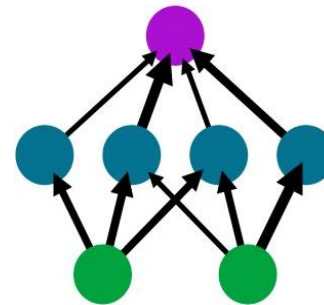
3) 'Neural Networks: optimization, processing layers

4) Neural-like' processing: DataSea

Four models of interaction: N-Nets

- 1) Follow Trees: pre-defined pull-downs (most app's)
- 2) Type our queries (e.g. search engines)

3) Neural Networks: optimization, processing layers
Pattern recognition, not database
Transformations
"Hidden layer" ... unpredictable



Output

*Calculations
(hidden-layer)*

Input

4) Neural-like' processing: DataSea

Four models of interaction: 'Neural-like'

- 1) Follow Trees: pre-defined pull-downs (most app's)
- 2) Type our queries (e.g. search engines)
- 3) Neural Networks: optimization, processing layers

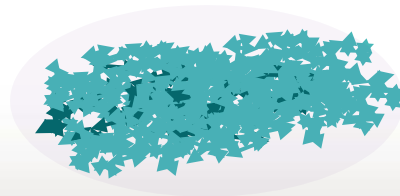
4) 'Neural-like' processing: DataSea

Data and Input --> neural-representation (fused, stateful)

Queries + commands + new info

Precise, explained answers

Context



Today's methods: Why so brittle?

Relational Databases: - fast, but not highly connected
Inferencing is hard!

“what is the **beast's** **name**?”

Table: Names

<u>Thing</u>	<u>Name</u>
robot	Gort
cat	Tally
planet	Earth
...	...

Today's methods: Query Expansion

What is the beast's (?cat's?) name?

Relational Databases: - fast, but not highly connected
Inferencing is hard!

“what is the **beast's** **name?**”

Table: Names

<u>Thing</u>	<u>Name</u>
robot	Gort
cat	Tally
planet	Earth
...	...

Table: Aliases

<u>Word</u>	<u>Alias</u>
dog	animal
dog	canine
cat	animal
cat	feline
animal	beast
...	...

```
graph LR; cat1[cat] --> animal1[animal]; cat2[cat] --> feline[feline]; animal2[animal] --> beast[beast];
```


Today's methods: Query Expansion

What is the beast's (?cat's?) name?

Relational Databases: - fast, but not highly connected
Inferencing is hard!

“what is the **beast's** **name?**”

Select Name from Names where Creature='beast'

Select Name ^{plus} from Names where Creature in
(select Word from Aliases where Alias='beast')

Select Name ^{plus} from Names where Creature in
(select Word from Aliases where Alias in
(select Word from Aliases where Alias='beast'))

Table: Names

<u>Thing</u>	<u>Name</u>
robot	Gort
cat	Tally
planet	Earth
...	...

Table: Aliases

<u>Word</u>	<u>Alias</u>
dog	animal
dog	canine
cat	animal
cat	feline
animal	beast
...	...

The logo for eComm 2008, featuring the word "eComm" in a stylized font with a swoosh over the "e", followed by "2008" in a bold, sans-serif font.

eComm™2008

Emerging Communications Conference

THE TRILLION DOLLAR INDUSTRY RETHINK™

“I know what I want,
why can't I just ask for it?”

DataSea's Query Expansion: automatic

“what is the beast's name?”

Tally, is Name

Associative Memory ... Single Network

Today's methods: Query Expansion

What is the beast's (cat's?) name?

Need to automatically:

- 1) Look further than one link
- 2) Treat 3 inferences in the same way as 1
- 3) Avoid 'query-explosion'

Fitting together ?

Object Reusability, Data integration ...

just connect them up?

```
Aaaaimport java.io.*;
import java.util.*;
import java.io.*;
import java.net.*;

import Utils.Login;
import Utils.Utils;
import net.sf.asterisk.manager.*;
import net.sf.asterisk.manager.event.*;
import net.sf.asterisk.manager.action.*;
import net.sf.asterisk.manager.impl.*;
import net.sf.asterisk.manager.response.*;
import net.sf.asterisk.fastagi.command.SelfPriorityCommand;
import net.sf.asterisk.fastagi.command.StreamFileCommand;

public class MGR_Reminder {
    static int DELAY_SLEEP_SECS = 3; // Writing from server to ast2goes about 40kB/sec, but to kvm ast
    is 500kB/sec

    public static void main(String[] argv) { // It's about 1MB per minute

    static public void writeFileToAsterisk(String filebase, String msg_str) {
        try {
            String command = utils.DS_ROOT+"/Eclipse/Scripts/ds_genAudToAst.sh "+user_name+"
            "+playback_str+" "+filebase+" "+msg_str;
            IExecCommand;
        } catch (java.io.IOException e) {
            System.err.println("Runtime IOException: " + e.toString());
        }
    }

    private DefaultManagerConnection getDefaultManagerConnection()
    {
        DefaultManagerConnection dmc;
        dmc = new DefaultManagerConnection();
        dmc.setUser("admin");
        dmc.setPassword("rock");
        return dmc;
    }

    public class CalcRate {
        public CalcRate {
            public CalcRate() {
                this.ASTERISK_MANAGER;
            }
        }
    }

    public Connection testConnection(String host, String db, String user, String pass) { // FATAL ERROR null dmc;
        String diag_str = "
        try {
            Class.forName("com.mysql.jdbc.Driver").newInstance();
        } catch (Exception ex) {
            System.err.println("ERROR in new instance of Login
            Class.forName("com.mysql.jdbc.Driver").newInstance(): "+ex);
            return ((Connection) null); // Bail out
        }

        Connection conn;
        try {
            com.mysql.jdbc.DriverManager.getDriver().getConnection("jdbc:mysql://" + host + "/" + db + "?user=" + user + "&password=" + pass);
        } catch (SQLException ex) {
            System.err.println("SQLException: " + ex.getMessage() + "\n");
            return ((Connection) null); // Bail out
        }

        if (conn == null)
            return (conn); // OK
    }

    public double sqlCalcRate(String phone_number) {
        int max_calls = 0;
        int max_calls = 100;
        String prefix = "
        String destname = "
        if (phone_number == null)
            return (-1);
        if (phone_number.length() < 3)
            return (DEFAULT_PHONE_COST_US_CANADA);
        if (phone_number.indexOf("01") != -1) // This is a US/Canada number
            return (DEFAULT_PHONE_COST_US_CANADA);
        else if (0 == phone_number.indexOf("011"))
            return (DEFAULT_PHONE_COST_US_CANADA);
        else
            phone_number = phone_number.substring(3); // Trim off the 01 and use the rest
            return (DEFAULT_PHONE_COST_US_CANADA);
        Connection conn = mysql_connection;
        if (conn == null)
            System.err.println("sqlCalcRate has null connections");
            return (-2);
        String query_str = "select Prefix, Rate, Rate from asteriskcdrdb.rates where Prefix like "
            + " " + phone_number.substring(0, 2) + "%";
        System.out.println("From sqlCalcRate
            + " phone_number = " + phone_number + "\n"
        }
    }
}
```

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.DriverManager;

public class CalcRate {
    public CalcRate {
        public CalcRate() {
            this.ASTERISK_MANAGER;
        }
    }

    public Connection testConnection(String host, String db, String user, String pass) { // FATAL ERROR null dmc;
        String diag_str = "
        try {
            Class.forName("com.mysql.jdbc.Driver").newInstance();
        } catch (Exception ex) {
            System.err.println("ERROR in new instance of Login
            Class.forName("com.mysql.jdbc.Driver").newInstance(): "+ex);
            return ((Connection) null); // Bail out
        }

        Connection conn;
        try {
            com.mysql.jdbc.DriverManager.getDriver().getConnection("jdbc:mysql://" + host + "/" + db + "?user=" + user + "&password=" + pass);
        } catch (SQLException ex) {
            System.err.println("SQLException: " + ex.getMessage() + "\n");
            return ((Connection) null); // Bail out
        }

        if (conn == null)
            return (conn); // OK
    }

    public double sqlCalcRate(String phone_number) {
        int max_calls = 0;
        int max_calls = 100;
        String prefix = "
        String destname = "
        if (phone_number == null)
            return (-1);
        if (phone_number.length() < 3)
            return (DEFAULT_PHONE_COST_US_CANADA);
        if (phone_number.indexOf("01") != -1) // This is a US/Canada number
            return (DEFAULT_PHONE_COST_US_CANADA);
        else if (0 == phone_number.indexOf("011"))
            return (DEFAULT_PHONE_COST_US_CANADA);
        else
            phone_number = phone_number.substring(3); // Trim off the 01 and use the rest
            return (DEFAULT_PHONE_COST_US_CANADA);
        Connection conn = mysql_connection;
        if (conn == null)
            System.err.println("sqlCalcRate has null connections");
            return (-2);
        String query_str = "select Prefix, Rate, Rate from asteriskcdrdb.rates where Prefix like "
            + " " + phone_number.substring(0, 2) + "%";
        System.out.println("From sqlCalcRate
            + " phone_number = " + phone_number + "\n"
        }
    }
}
```

Fitting together ?

Object Reusability,
Data integration ...

just connect them up?

*rather
complex*

```
AAAAimport java.io.*;
import java.util.*;
import java.io.*;
import java.net.*;

import Utils.Login;
import Utils.Utils;
import net.sf.asterisk.manager.*;
import net.sf.asterisk.manager.event.*;
import net.sf.asterisk.manager.action.*;
import net.sf.asterisk.manager.impl.*;
import net.sf.asterisk.manager.response.*;
import net.sf.asterisk.fastagi.command.SelfPriorityCommand;
import net.sf.asterisk.fastagi.command.StreamFileCommand;

public class MGR_Reminder {
    static int DELAY_SLEEP_SECS = 3; // Writing from server to ast2goes about 40kB/sec, but to kvm ast
    is 500kB/sec

    public static void main(String[] argv) { // It's about 1MB per minute

        static public void writeFileToAsterisk(String filebase, String msg_str) {
            try {
                String command = Utils.DS_ROOT+"/Eclipse/Scripts/ds_genAudToAst.sh "+user_name+"
                "+playback_str+" "+filebase+" "+msg_str;
                it.executeCommand();
            } catch (java.io.IOException e) {
                System.err.println("Runtime IOException: " + e.toString());
            }
        }

        private DefaultManagerConnection getDefaultManagerConnection()
    }
}
```

```
import java.sql.Connection;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.DriverManager;
import java.sql.Driver;

public class CalcRate {
    public CalcRate() {
        this.mysql_connection = null;
    }

    public Connection testConnection(String host, String db, String user, String pass) {
        String cmd_str = "mysql -h"+host+" -u"+user+" -p"+pass+" "+db+" >>>";
        try {
            Class.forName("com.mysql.jdbc.Driver").newInstance();
        } catch (Exception e) {
            System.err.println("ERROR in new Instance of Login");
        }
        try {
            Class.forName("com.mysql.jdbc.Driver").newInstance();
        } catch (Exception e) {
            System.err.println("ERROR in new Instance of Login");
        }
        return ((Connection) null); // Bail out
    }

    Connection conn;
    try {
        conn = DriverManager.getConnection("jdbc:mysql://" + host + "/" + db + "?user=" + user + "&password=" + pass);
    } catch (SQLException e) {
        System.err.println("ERROR testConnection SQLException: " + e.getMessage() + "\n");
        return ((Connection) null); // Bail out
    }
    if (conn == null) {
        return (conn); // OK
    }
}

public double sqlCalcRate(String phone_number) {
    String prefix;
    int i;
    String rate;
    String destname;
    if (phone_number == null)
        return 0.0;
    if (phone_number.length() < 3)
        return 0.0;
    if (0 == phone_number.indexOf("1")) // This is a US/Canada number
        return 0.0;
    else if (0 == phone_number.indexOf("011"))
        return 0.0;
    else
        phone_number = phone_number.substring(3); // Trim off the 01 and use the rest
    Connection conn = mysql_connection;
    if (conn == null)
        System.err.println("sqlCalcRate has null connections");
    return (-2);
}

String query_str = "select Prefix, Rate, Rate from asteriskcdrdb.rates where Prefix like "
+ phone_number + "%";
System.err.println("sqlCalcRate query: " + query_str);
System.err.println("sqlCalcRate result: " + rate);
}
```

Fitting together ?

Object Reusability,
Data integration ...

& Brain Transplants:

just connect them up?

*rather
complex*

```
AAAAimport java.io.*;
import java.util.*;
import java.awt.*;
import java.net.*;

import Utils.Login;
import Utils.Utils;
import net.sf.asterisk.manager.*;
import net.sf.asterisk.manager.event.*;
import net.sf.asterisk.manager.action.*;
import net.sf.asterisk.manager.impl.*;
import net.sf.asterisk.manager.response.*;
import net.sf.asterisk.fastagi.command.SelfPriorityCommand;
import net.sf.asterisk.fastagi.command.StreamFileCommand;

public class MGR_Reminder {
    static int DELAY_SLEEP_SECS = 3; // Writing from server to ast2goes about 40kB/sec, but to kvm ast
    is 500kB/sec

    public static void main(String[] argv) { // It's about 1MB per minute

    static public void writeFileToAsterisk(String filebase, String msg_str) {
        try {
            String command = Utils.DS_ROOT+"/Eclipse/Scripts/ds.genAudToAst.sh "+user_name+"
            "+playback_str+" "+filebase+" "+msg_str;
            it.execCommand();
        } catch (java.io.IOException e) {
            System.err.println("Runtime IOException: " + e.toString());
        }
    }

    private DefaultManagerConnection getDefaultManagerConnection()
}
```

```
import java.sql.Connection;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.DriverManager;
import java.sql.Driver;

public class CalcRate {
    public CalcRate() {
        this.mysql_connection = null;
    }

    public Connection testConnection(String host, String db, String user, String pass) {
        String cmd_str = "mysql -h"+host+" -u"+user+" -p"+pass+" "+db+" >>>";
        try {
            Class.forName("com.mysql.jdbc.Driver").newInstance();
        } catch (Exception e) {
            System.err.println("ERROR in new Instance of Login");
            return ((Connection) null); // Bail out
        }
        try {
            Connection conn = DriverManager.getConnection("jdbc:mysql://" + host + "/" + db + "?user=" + user + "&password=" + pass);
        } catch (SQLException e) {
            System.err.println("ERROR testConnection SQLException: " + e.getMessage() + "\n");
            return ((Connection) null); // Bail out
        }
        if (conn == null)
            return ((Connection) null); // OK
    }

    public double sqlCalcRate(String phone_number) {
        int i;
        int i_max = 100;
        String prefix = "00";
        String destname = "";
        if (phone_number == null)
            return 0.0;
        if (phone_number.length() < 3)
            return 0.0;
        if (phone_number.indexOf("1") != -1) // This is a US/Canada number
            return 0.0;
        else if (phone_number.indexOf("011") != -1)
            return 0.0;
        else
            phone_number = phone_number.substring(3); // Trim off the 01 and use the rest
        Connection conn = mysql_connection;
        if (conn == null)
            System.err.println("sqlCalcRate has null connections");
        return (-2);
        String query_str = "select Prefix, Rate, Rate from asteriskcdrdb.rates where Prefix like "
        + "00%";
        System.err.println("sqlCalcRate has query: "
        + query_str);
        return 0.0;
    }
}
```

Fitting together ?

Biology

*Immune system,
Protein 'lock and key' model*

*rather
complex*

Fitting together ?

Biology

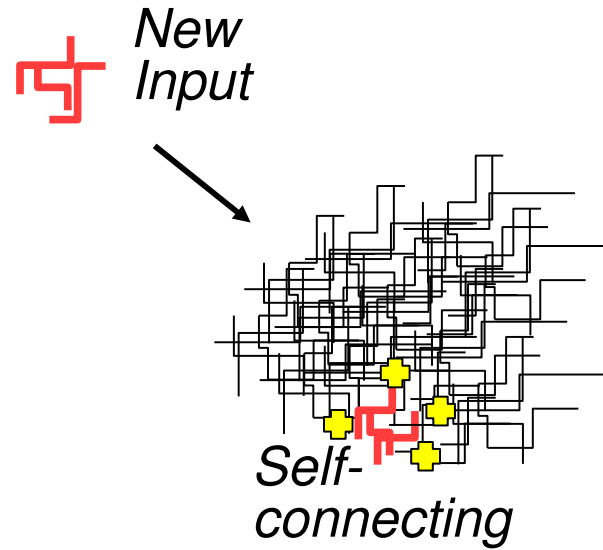
*Immune system,
Protein 'lock and key' model*

*complex ...
But biological systems are self organizing*

Self-organizing => don't need to know details

Self-organizing: The Solution?

*Self-organizing
data*



Self-organizing => don't need to know details

Self-organizing: The Solution?

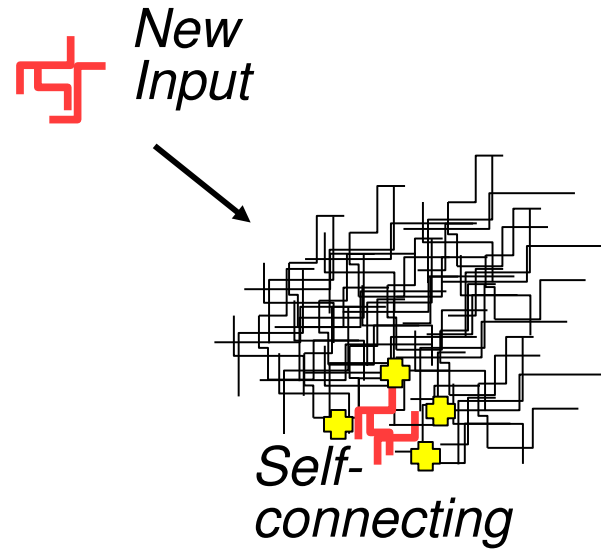
Are Operations different from Data?

Self-organizing: The Solution?

*Are Operations different from Data?
Data & Operations merge*

Self-organizing: The Solution?

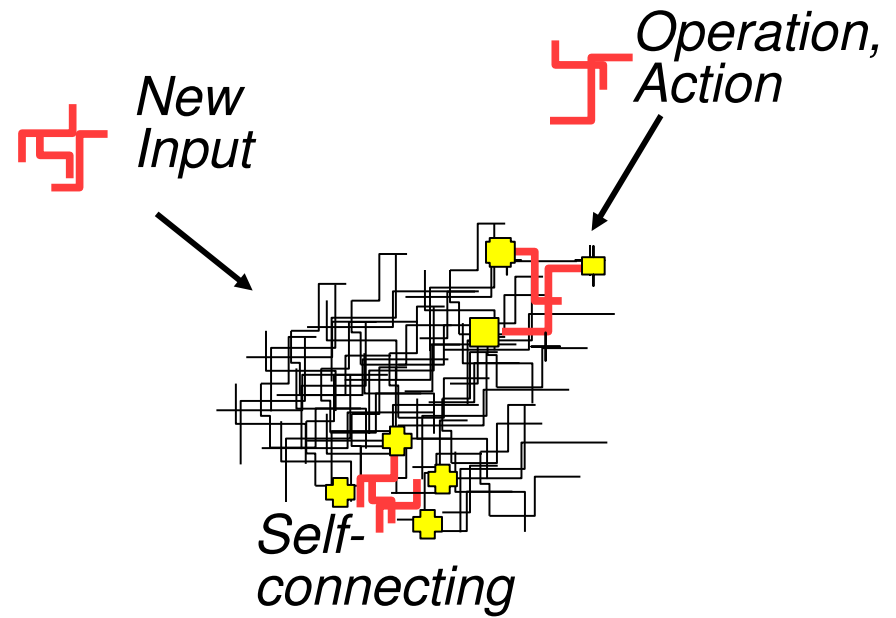
*Self-organizing
data and operations*



Self-organizing => don't need to know details

Self-organizing: The Solution?

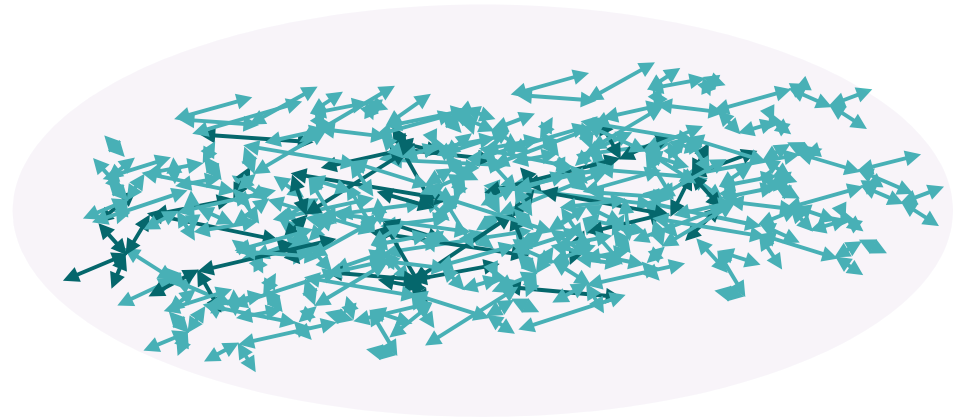
*Self-organizing
data and functions*



Self-organizing => don't need to know details

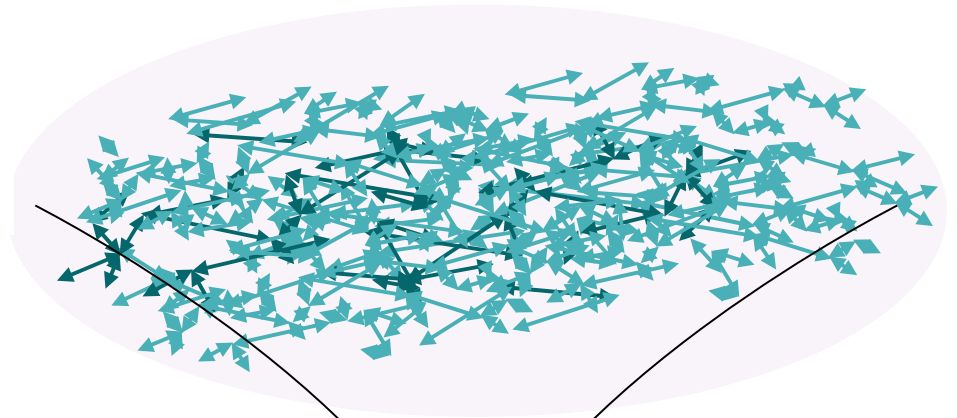
DataSea Network

Neural-like network of fused data



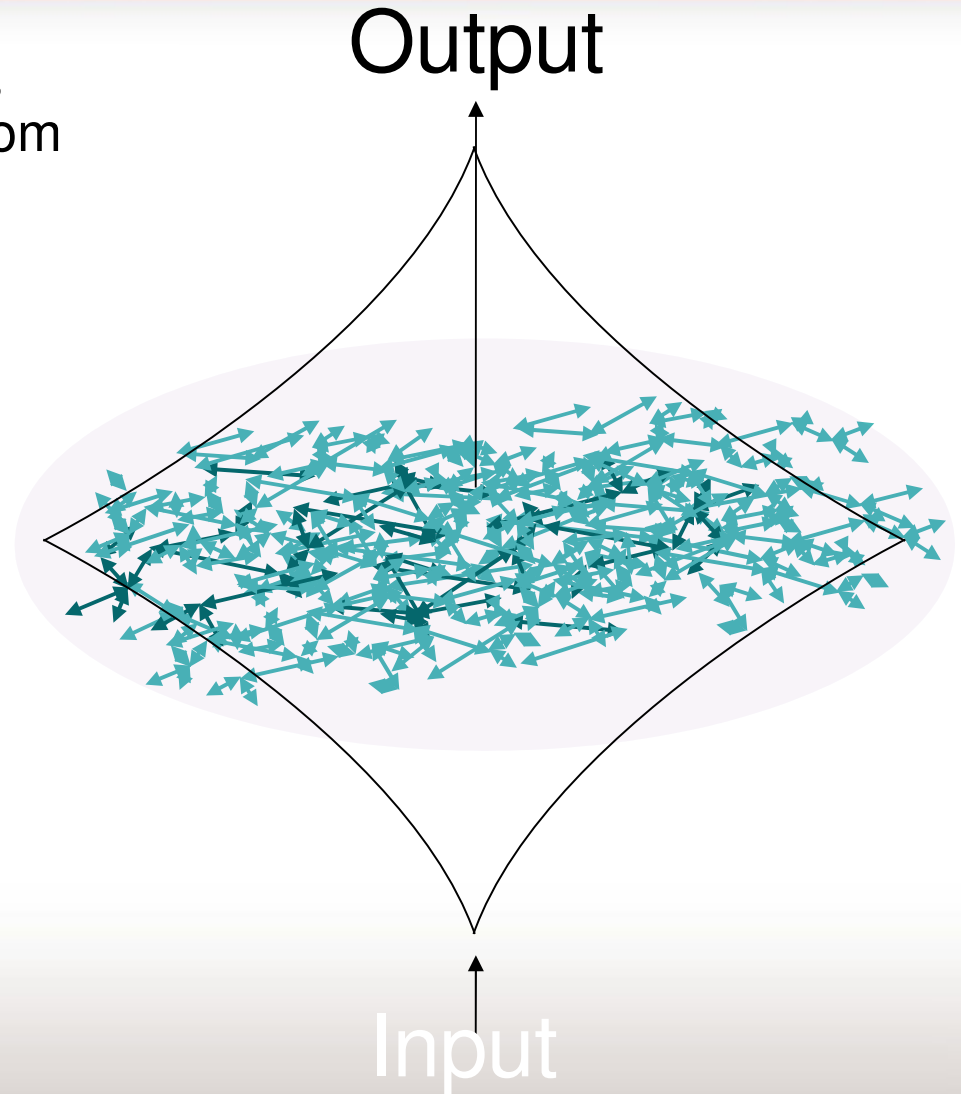
DataSea Network

Processing ...



DataSea Network

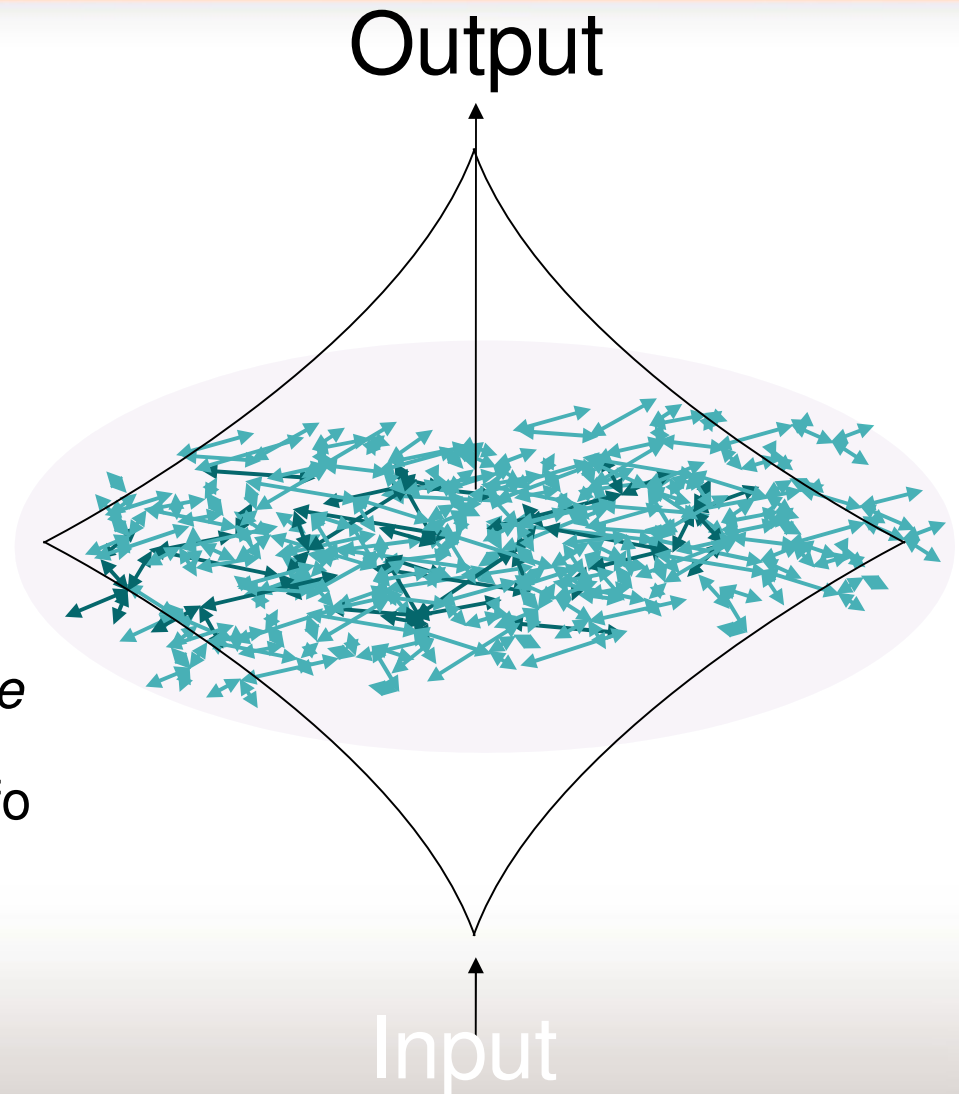
Answers
pulled from
network



DataSea Network

Fusion ...

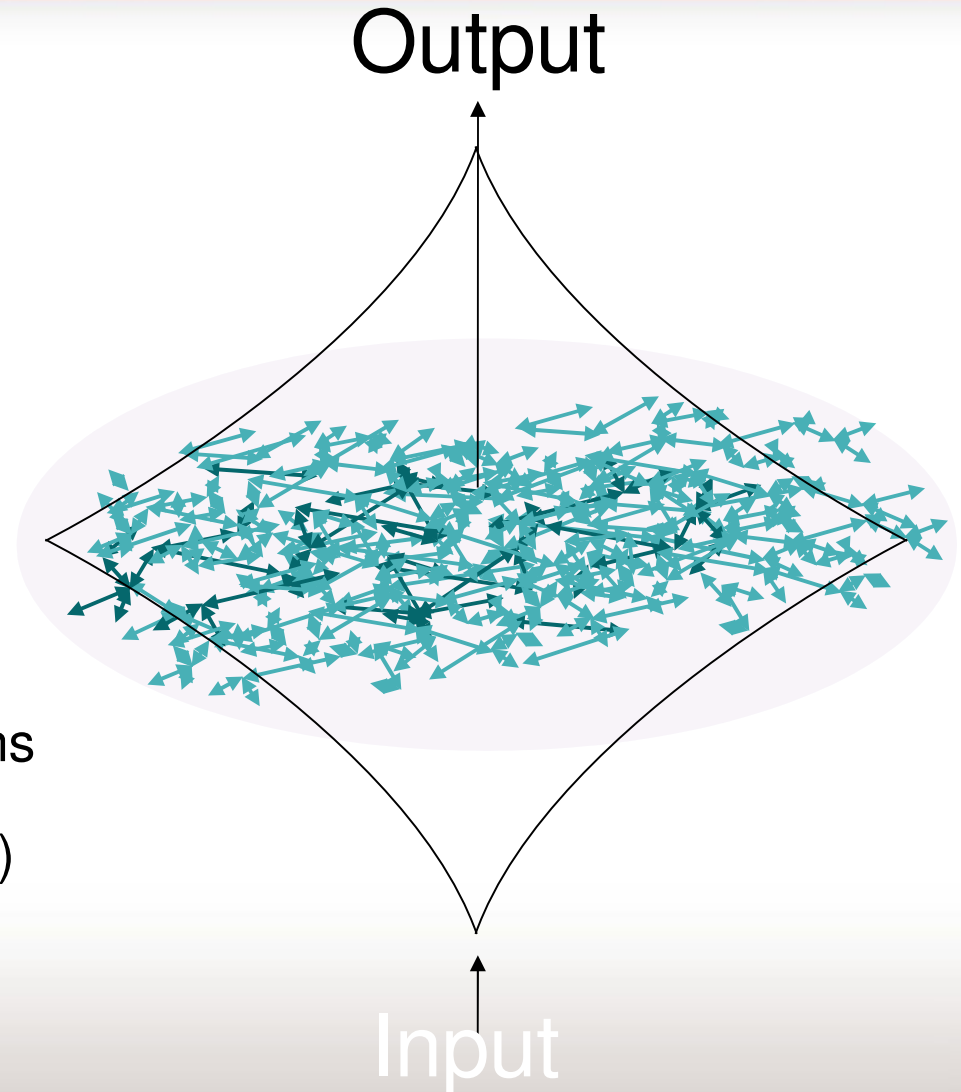
- Entire network accessible
you can get there from here
- Immediate influence from new info
- Inferencing



DataSea Network

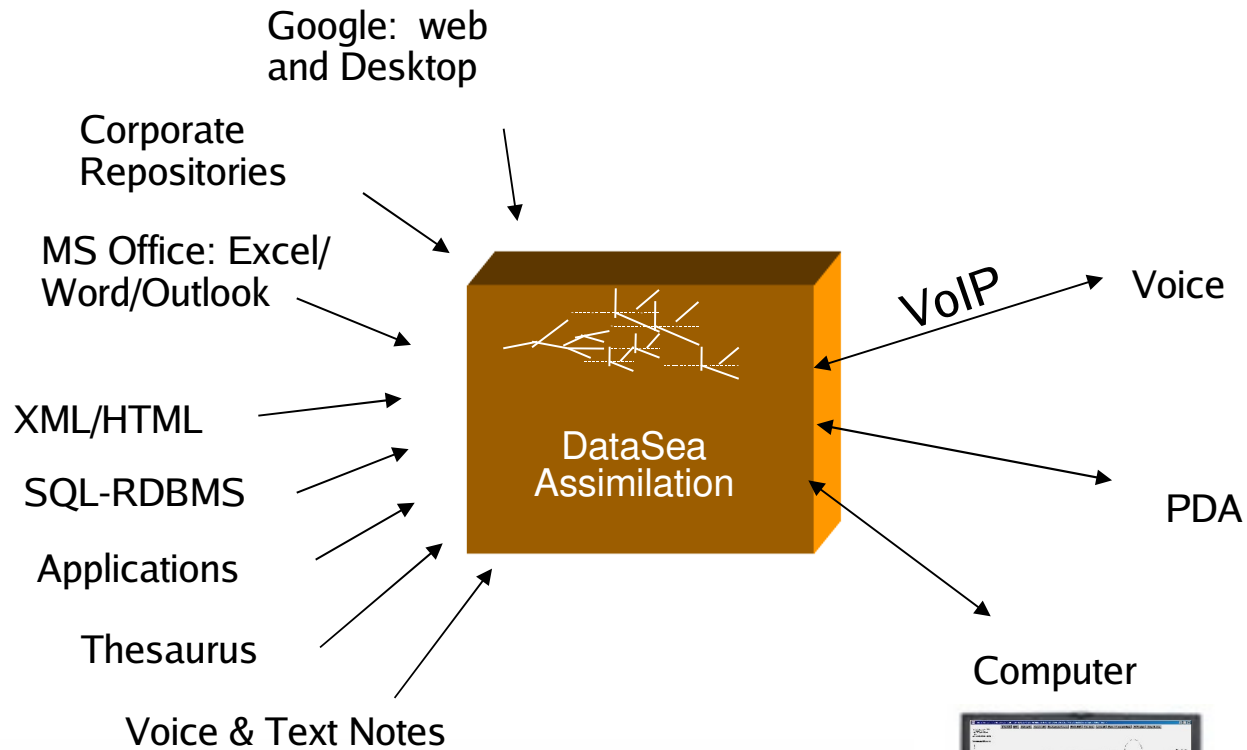
Reduces ...

- difficult UI navigation
- multiple, precise steps & decisions
- False negatives (null intersection)



Multi-Source, Multi-modal :

Inputs

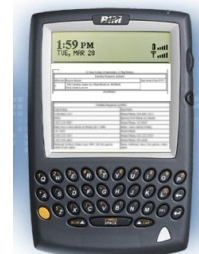


Outputs

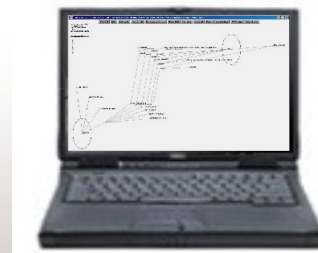
VoIP → Voice



→ PDA



→ Computer



Tricky Parts Solved

How to:

Represent data and functions

→ self-organizing, neural-like network

Process the network

→ simple algorithms

Package answers

→ extract activated elements, reassemble

Integrating with the Real World

Upload Files

AddressBook, SpreadSheets, Text, HTML

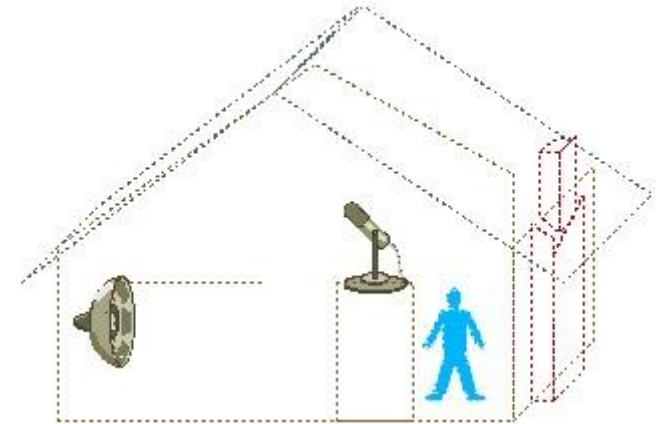
Crawl Relational DataBase

Customize GUI for application specific needs

Home Companion (*prototype*)

Typed inputs work ==> Voice (w/ VR) works ...

- Microphone & speakers in house
- DataSea behind VR
- Example
 - Audrey! (starts listening)
yes? ...
 - What is Linda's cell?"
'Linda's mobile is (604)9xx-yyyy
 - Call her and say Hi, I'll be late for the meeting.
'calling (604)9xx-yyyy with the message 'Hi, I'll be late...'
 - Thank you
you're welcome. (stops listening)



Point of View ...

Data Modules

Load Another User's Data, run a query:

→ see the world *from their Point of View*

Application ...

Product 1:

Salesforce.com plugin (beta)

Gives:

- Single-Step
- Query expansion
- Actions

SF: Current method: failures, # steps Finding accounts and contacts in Chicago

- 1) Search, Chicago → fails ... null answer
- 2) Click Advanced Search,
- 3) Select Find All,
- 4) Type: Chicago

Search All

Go! | own

Limit Search | own

Advanced Search...

Create New...

- Recent Items
- [Ultimate Co.](#)
 - [00001046](#)
 - [Ursula Anderson](#)
 - [Dave Wilson](#)
 - [John Johnson](#)
 - [Steve Johnson](#)
 - [00001027](#)
 - [Steve Johnson](#)
 - [Steve Smalley](#)
 - [Stupendous Stuff](#)



Search Limit to items I own

Scope: All Objects | [Advanced Search](#)

[Accounts \[1\]](#) | [Contacts \[2\]](#)

Accounts [1]

Admins: Search results filters can help your users find their data faster. Click 'Customize' to configure them for your organization!
[Hide this message for: Accounts](#) | [All Objects](#)

Action	Account Name	Account Site	Phone	Owner Alias
Edit	Grand Hotels & Resorts Ltd		(312) 596-1000	LWebs

Contacts [2]

Admins: Search results filters can help your users find their data faster. Click 'Customize' to configure them for your organization!
[Hide this message for: Contacts](#) | [All Objects](#)



Action	Name	Account Name	Account Site	Phone	Email	Contact Owner Alias
Edit	Mr. Tim Barr	Grand Hotels & Resorts Ltd		(312) 596-1000	barr_tim@grandhotels.com	LWebs
Edit	Mr. John Bond	Grand Hotels & Resorts Ltd		(312) 596-1000	bond_john@grandhotels.com	LWebs

There are no matching: Leads, Opportunities, Cases, Case Comments, Campaigns, Contracts, Assets, Tasks and Events, Notes, Attachments, Reports or Users

SF: DataSea; one step, richer response

Finding accounts and contacts in Chicago












1) "show Chicago"


Setup · System Log · Help · Logout  Sales

[Home](#)
[DataSea](#)
[Campaigns](#)
[Leads](#)
[Accounts](#)
[Contacts](#)
[Opportunities](#)
[Forecasts](#)
[Contracts](#)
[Cases](#)
[Solutions](#)
[Products](#)
[Reports](#)
[Documents](#)
[Dashboards](#)

Search
 Search All

 Limit to items I own
[Advanced Search...](#)

Recent Items
 [Ultimate Co.](#)
 [00001046](#)
 [Ursula Anderson](#)
 [Dave Wilson](#)
 [John Johnson](#)
 [Steve Johnson](#)
 [00001027](#)
 [Steve Johnson](#)
 [Steve Smalley](#)
 [Stupendous Stuff](#)
 [Recycle Bin](#)

PRIOR-INPUT: SHOW CHICAGO

0017000000MmkTdAAJ, is Id

Condensed Answers

ACCOUNT (Company)	CONTACT (Name)	OWNER	CASE	TASK	EVENT	SOL'N	LEAD	OPP'Y
Grand Hotels & Resort...								
Grand Hotels & Resort...	John Bond	*Linda Webster	*Delay in installation...					
Grand Hotels & Resort...	Tim Barr	*Linda Webster	*Customer service for ...					
Grand Hotels & Resort...		*Linda Webster						Grand Hotels SLA
Grand Hotels & Resort...								Grand Hotels Kitchen ...
Grand Hotels & Resort...								
Grand Hotels & Resort...								
Grand Hotels & Resort...								
Grand Hotels & Resort...								
Grand Hotels & Resort...								Grand Hotels Guest Po...
Grand Hotels & Resort...								Grand Hotels Generato...
Grand Hotels & Resort...								Grand Hotels Emergenc...
*Grand Hotel/ Resort...								
*Grand Hotel/ Resort...								
*Grand Hotel/ Resort...								
*Grand Hotel/ Resort...								

Account

Name: Grand Hotels & Resorts Ltd

Description: Chain of hotels and resorts across the US, UK, Eastern Europe, Japan, and SE Asia.

AccountNumber: CD439877

Phone: (312) 596-1000

Website: www.grandhotels.com

Industry: Hospitality

NumberOfEmployees: 5600

BillingStreet: --

BillingCity: Chicago

BillingPostalCode: --

BillingCountry: --

ShippingStreet: --

ShippingCity: --

ShippingPostalCode: --

ShippingCountry: --

[Add:Event/Reminder](#) [Add:Case](#)

SF: Current method

Calling the contact for Grand Hotels

- 1) Search,
 - 2) Detail Page,
 - 3) contacts-DetailPage,
 - 4) place call: dial, wait, speak
- Time: 2+ minutes

The screenshot displays the Salesforce user interface. At the top, there are navigation menus for 'Setup', 'System Log', 'Help', and 'Logout', along with an 'AppExchange' dropdown set to 'Sales'. The main navigation bar includes tabs for 'Home', 'DataSea', 'Campaigns', 'Leads', 'Accounts', 'Contacts', 'Opportunities', 'Forecasts', 'Contracts', 'Cases', 'Solutions', 'Products', 'Reports', 'Documents', and 'Dashboards'. The 'Accounts' tab is currently selected.

On the left sidebar, there is a search bar with 'Accounts' selected and a 'Go!' button. Below it, there are options for 'Limit to items I own' and 'Advanced Search...'. A 'Create New...' dropdown is also visible. The 'Recent Items' section lists several users and accounts, including 'Jane Grey', 'Grand Hotels & Resorts Ltd', 'Ultimate Co.', and '00001046'.

The main content area shows search results for 'Grand Hotels'. The search scope is set to 'Accounts' and 'Advanced'. Below the search bar, there are filters for 'Accounts [1]' and 'Admins: Search results filters'. A table lists the search results with columns for 'Action' and 'Account Name'. The first result is 'Grand Hotels & Resorts Ltd'.

The detailed view for 'Grand Hotels & Resorts Ltd' is shown. It includes a header with the account name and links for 'Printable View', 'Customize Page', and 'Help for this Page'. Below the header, there are tabs for 'Contacts [2]', 'Opportunities [5]', 'Cases [4]', 'Open Activities [0]', 'Activity History [0]', 'Notes & Attachments [0]', and 'Partners [0]'. The 'Contacts' tab is active, showing a table of contacts with columns for 'Action', 'Contact Name', 'Title', 'Email', and 'Phone'. Two contacts are listed: 'Tim Barr' (SVP, Administration and Finance) and 'John Bond' (VP, Facilities).

Below the contacts table, there is a detailed view of the account information, including 'Account Site', 'Type', 'Industry', 'Annual Revenue', 'Billing Address', 'Customer Priority', 'SLA Expiration Date', 'Number of Locations', 'Active' status, 'Created By', 'Description', 'Ticker Symbol', 'Ownership', 'Employees', 'SIC Code', 'Shipping Address', 'SLA', 'SLA Serial Number', 'Upsell Opportunity', and 'Last Modified By'.

Action	Contact Name	Title	Email	Phone
Edit Del	Tim Barr	SVP, Administration and Finance	barr_tim@grandhotels.com	(312) 596-1000
Edit Del	John Bond	VP, Facilities	bond_john@grandhotels.com	(312) 596-1000

Account Site	Ticker Symbol	GHTL
Type	Ownership	Public
Industry	Employees	5,600
Annual Revenue	SIC Code	2268
Billing Address	Shipping Address	2334 N. Michigan Avenue, Suite 1500 Chicago, IL 60601, USA
Customer Priority	SLA	Gold
SLA Expiration Date	SLA Serial Number	5572
Number of Locations	Upsell Opportunity	Yes
Active		Yes
Created By	Last Modified By	Linda Webster, 12/18/2007 11:08 AM
Description		Chain of hotels and resorts across the US, UK, Eastern Europe, Japan, and SE Asia.

SF: DataSea's method

Calling the contact for Grand Hotels

- 1) "show Grand Hotels"
 - 2) "call Tim Barr and say Hi Tim, I'll be there at noon, sorry for delay."
- Time: 0.5 mins


Setup · System Log · Help · Logout
AppExchange
Sales

Home
DataSea
Campaigns
Leads
Accounts
Contacts
Opportunities
Forecasts
Contracts
Cases
Solutions
Products
Reports
Documents
Dashboards

Search

Search All

Go!

Limit to items I own

Advanced Search...

Create New...

Recent Items

- Ultimate Co.
- 00001046
- Ursula Anderson
- Dave Wilson
- John Johnson
- Steve Johnson
- 00001027
- Steve Johnson
- Steve Smalley
- Stupendous Stuff

Recycle Bin

DATASEA

Ready: User=lwebster_sf@datasea.com

Completed: 'call Tim Barr' (3.717 secs)

Awake for 1 hour, 2 mins, as of: Feb 14, 2008 8:44:34 PM.

Input ...

Word-hints:

PRIOR-INPUT: CALL TIM BARR

Say 'execute' to _CALL_ 312-596-1000 [Phone]. The message is: *Hi Tim, I'll be there at noon, sorry for the delay..*

Costs:(US/Canada) is \$0.025 per minute, your phone (6049885875, US/Canada) is \$0.025 per minute.Your current balance is \$0.8.

312-596-1000, is Phone

Condensed Answers

ACCOUNT (Company)	CONTACT (Name)	OWNER	CASE	TASK	EVENT	SOL'N	LEAD	OPP'Y
*Grand Hotels & Resort...	Tim Barr	*Linda Webster	*Customer service for ...					
*Grand Hotels & Resort...	Tim Barr	*Linda Webster	Structural breakdown ...	*Customer service for ...				

Execute _CALL_ 312-596-1000 [Phone]

SF-CONTACT

Broad Applicability

Applications:

- Salesforce.com (beta)
- Personal Assistant
- Corporate DB/Knowledge-base
- Corporate web portal
- Identity Sharing
- Home Voice companion (prototyped)
- Business Portal (design)
- Intelligence community



Broad Applicability

Applications:

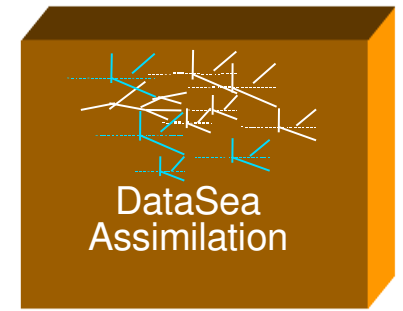
- Salesforce.com (beta)
- Personal Assistant
- Corporate DB/Knowledge-base
- Corporate web portal
- Identity Sharing
- Home Voice companion (prototyped)
- Business Portal (design)
- Intelligence community



Broad Applicability

Applications:

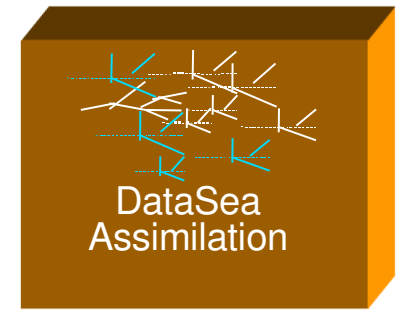
- Salesforce.com (beta) →
- Personal Assistant →
- Corporate DB/Knowledge-base →
- Corporate web portal →
- Identity Sharing
- Home Voice companion (prototyped)
- Business Portal (design)
- Intelligence community



Broad Applicability

Applications:

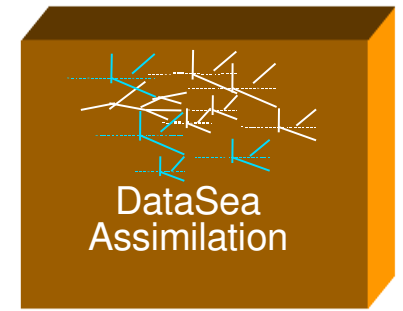
- Salesforce.com (beta) →
- Personal Assistant →
- Corporate DB/Knowledge-base →
- Corporate web portal →
- **Identity Sharing** →
- Home Voice companion (prototyped)
- Business Portal (design)
- Intelligence community



Broad Applicability

Applications:

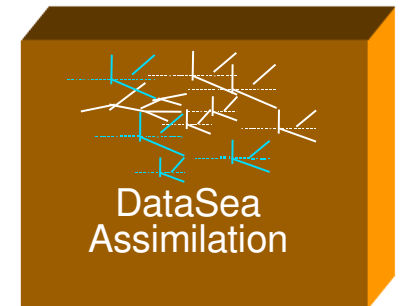
- Salesforce.com (beta) →
- Personal Assistant →
- Corporate DB/Knowledge-base →
- Corporate web portal →
- Identity Sharing →
- Home Voice companion (prototyped) →
- Business Portal (design)
- Intelligence community



Broad Applicability

Applications:

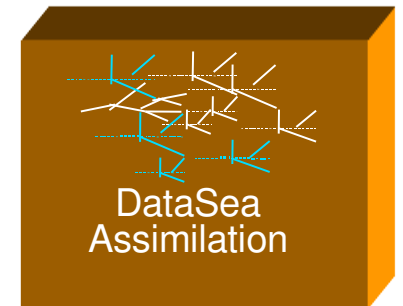
- Salesforce.com (beta) →
- Personal Assistant →
- Corporate DB/Knowledge-base →
- Corporate web portal →
- Identity Sharing →
- Home Voice companion (prototyped) →
- Business Portal (design) →
- Intelligence community



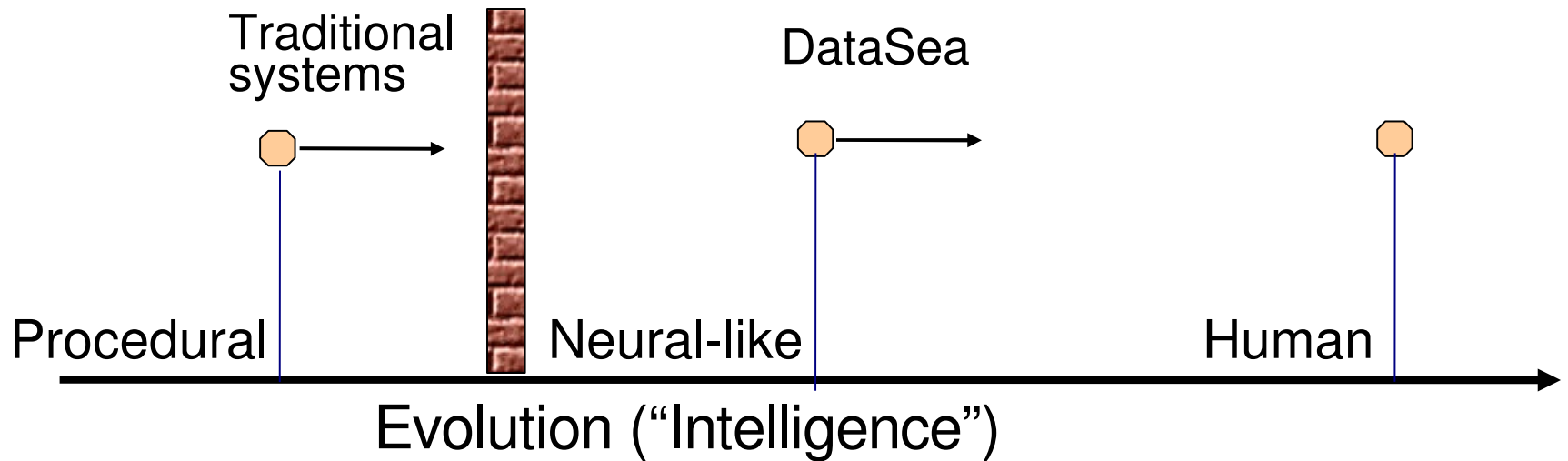
Broad Applicability

Applications:

- Salesforce.com (beta) →
- Personal Assistant →
- Corporate DB/Knowledge-base →
- Corporate web portal →
- Identity Sharing →
- Home Voice companion (prototyped) →
- Business Portal (design) →
- Intelligence community →

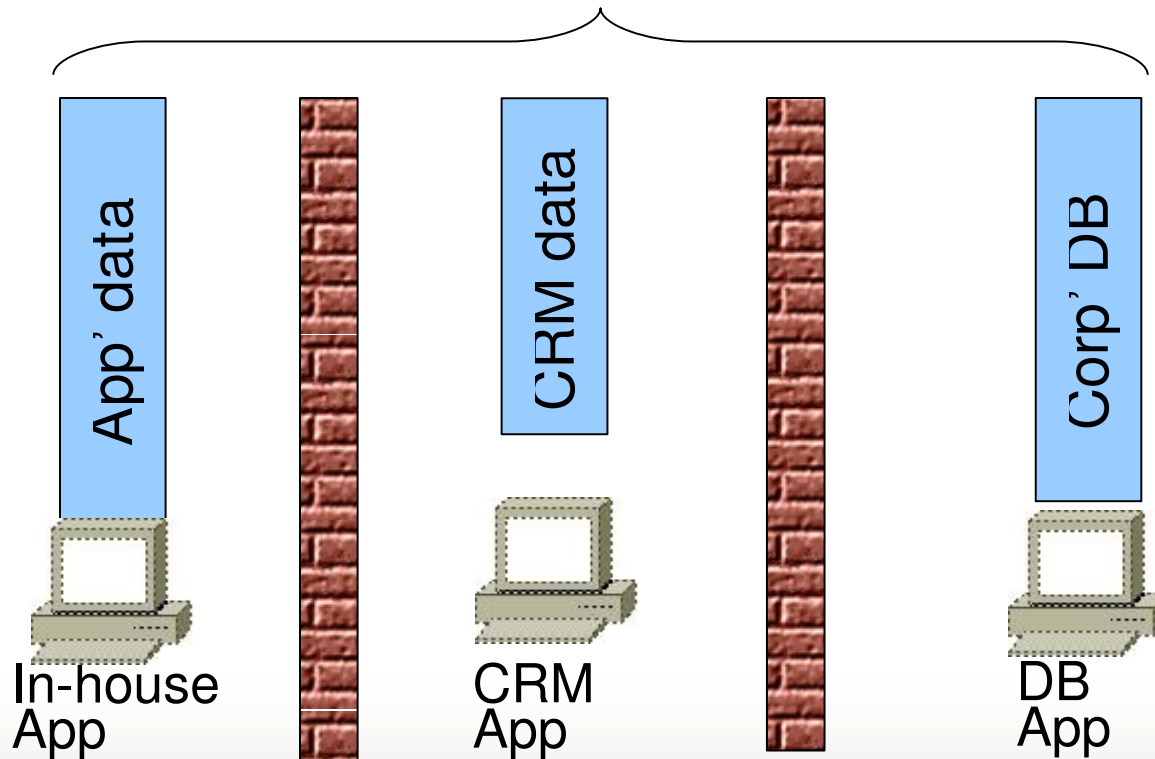


Evolution of Search and Actions



Trans-Application Fusion

DataSea Assimilates,
'connects the dots'



Summary

- Technology:
 - Highly-connected
 - Fused, cross-domain data
 - Self-organizing

- Results
 - Inferencing & query expansion
 - Natural Language I/O
 - Plugin data ('knowledge modules') and functions

Contact Us

DataSea, Inc.
www.DataSea.com

CEO: Rocky Nevin
CFO: Linda Webster

RNevin@DataSea.com
LWebster@DataSea.com

SalesForce DataSea Plugin
Pending SFDC Appexchange certification