



Building on the Survox Platform

2015 Survox Summit

Building on the Survox Platform



Why is this important to you?

- The Survox platform integrates into your data streams and workflows
- Speeds both data collection and data analysis to shorten time-to-insights

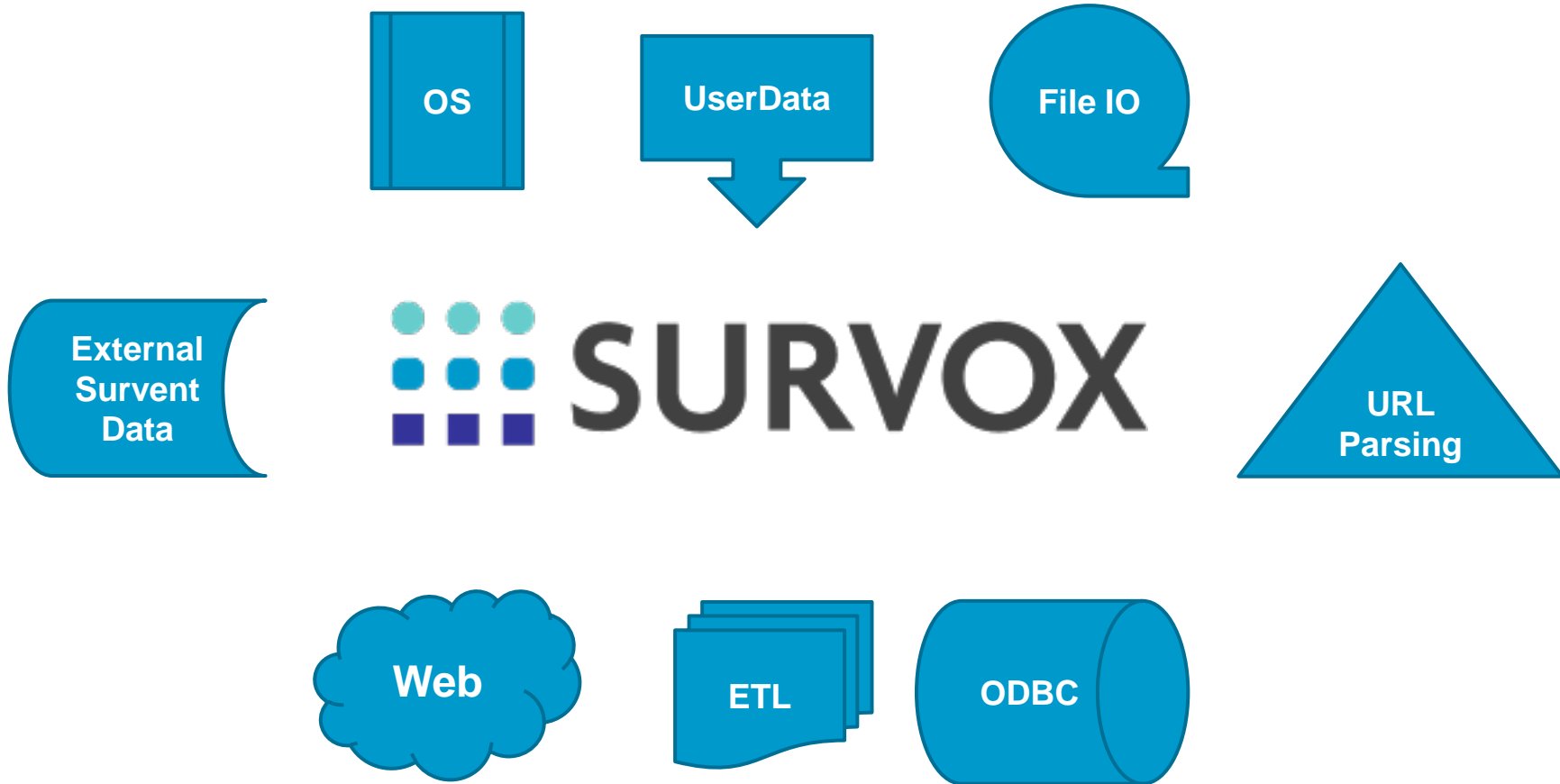
Agenda

- Overview methods of integration
- Review examples and see how other customers have extended the Survox platform to solve their research designs.

Presenters

- Josh Smith, Professional Services Developer
- Allen Porter, Director of Customer Success

Survox Platform Integrations

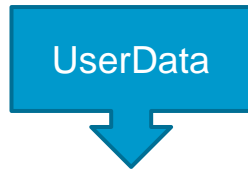


A background network diagram with blue lines and nodes. The nodes are represented by circles and squares of varying sizes, connected by thin lines. The overall color scheme is light blue and white.

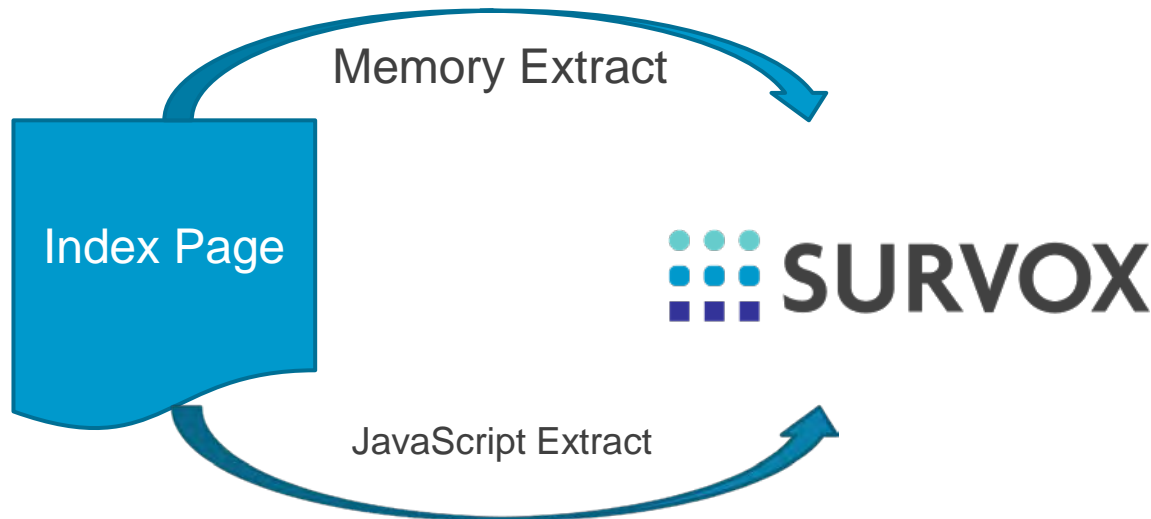
URL Data

JavaScript Parsing, PHP, UserData

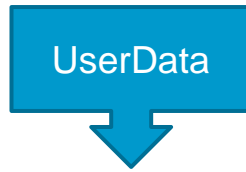
UserData



- Allows you to pass data into Survent from your index page
- All server-side
- Pull into survey from local scratch
- 120 bytes max
- PHP let's you create data, extract from URL, interact with JavaScript or respondent



UserData



Example:

```
index.php
```

```
<input type="hidden" id='USER_DATA' name="USER_DATA"  
value="<?php echo $deviceType; ?>" />
```

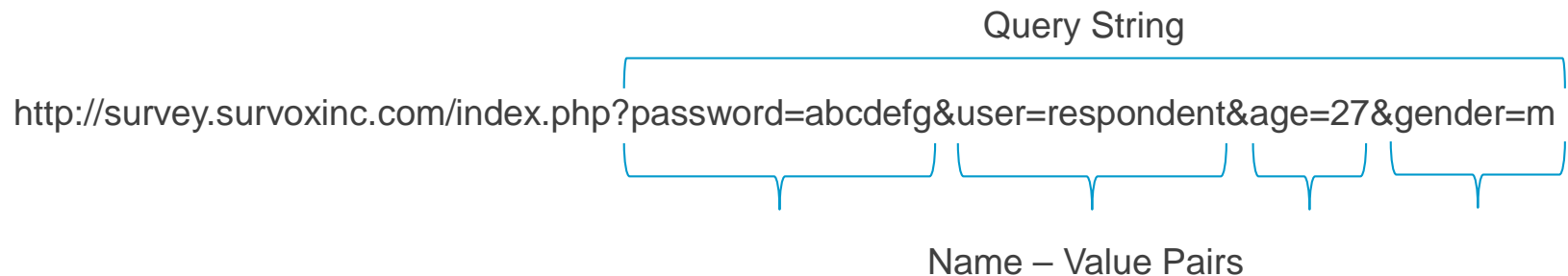
```
qpx
```

```
' ' from the USER_DATA hidden input  
{ NAV_RESDEVICE: [DEVICE.6]  
!SPECIAL, FROM_LOCALSCRATCH_to_data , DEVICE, 1, 6 }
```

URL Variables

URL
Parsing

- Information passed through the url itself
- This is an automatic function of index.html so you only need to set the script up in your spec to gather and store this data.
- This method will only work on the first page of the survey but you can pass approximately 2000 columns of information (depending on the browser type).
- Can parse URL by name or location



URL Variables

URL
Parsing

```
?name=<username>&password=<password>&respname=Charlie&age=45&gender=1;other=ab
```

```
{respname:  
!var,h,40,0}
```

```
{age:  
!var,h,2,0}
```

```
{gender:  
!var,h,1,0}
```

```
{other:  
!var,h,2,0}
```

```
{  
<script type="text/javascript">
```

```
var get_by = "name"; //set as location or name.
```

```
var urllist = new Array(  
"respname", "age", "gender", "other");
```

```
var varlist = new Array(  
"respname", "age", "gender", "other");
```

```
parse_query();  
</script>  
!DISP }
```




- PHP can parse URLs
- `Parse_str()` exploded query string into PHP variables
- Server side
- Validation for security
- Combine with hashing and encryption for advanced security

URL Variables

URL
Parsing

```
<?php
//$qstring = "zUR87er2FghYOTRjOTM204OGJmLTQ5YzItYT94oFtGnNjAyNTY1MDc1=";

$qstring = $_SERVER['QUERY_STRING'];

//incase there are other name-value pairs appended to the URL, just strip
them off
if ( strpos($qstring, '&') > 0 ) {
    $qstring = strstr($qstring, '&', true);
}

//put the shared secret in a separate file with locked permissions
//use include file for production
$shared_secret = "HelloWorld";

//decode the entire query string
$qstring_d64 = base64_decode($qstring);
```

URL Variables

URL
Parsing

```
// last 32 bytes is the check sum so TotalLen - 32 = HashLen
//grab everything but the last 32 bytes
$clear_text_str = substr($qstring_d64, 0, -32);
//grab the last 32 bytes
$checksum1 = substr($qstring_d64, -32);

//calculate the checksum ourselves with the shared secret
$checksum2 = md5($clear_text_str.$shared_secret);

//upshift because md5 comes back lower case
if ( strtoupper($checksum1) <> strtoupper($checksum2) ) {
    //log this event
    $log = fopen('login_hitlog.txt', 'a');
    $currtime = date("YmdHis");
    fwrite($log,
$currtime."|". "failurldecode."|" . $qstring."|_|_|_|_|_|". $_SERVER['REMOTE_ADDR']."\n");
    die("<br><br>The survey link is bad.\n");
} else {
    //now explode the good query string
    parse_str($clear_text_str); //explode the query string into variables
}
```

URL Variables

URL
Parsing

```
//validate all the key-value pairs
//preg match returns boolean t/f
//Example of Time Stamp Check
//      preg_match("/^20[0,1-9]{2}[-][0,1-9]{2}[-][0,1-9]{2}[T][0,1-9]{2}[:][0,1-9]{2}[:][0,1-9]{2}$/", $DateTime)

?>
```

Bringing it all together

URL
Parsing

Please enter your values to generate a query string.

Age: ex: 33

Store ID: ex: 001

City: ex: Philadelphia

Submit Form

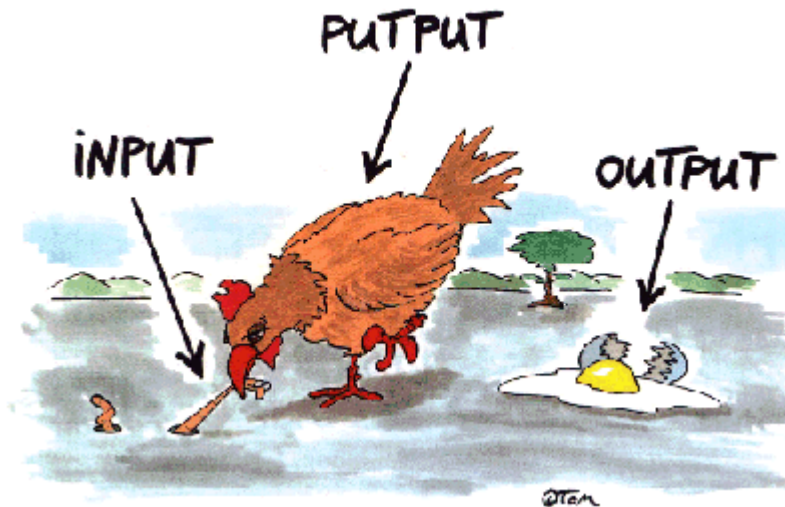
The background of the slide features a network diagram with various nodes and connecting lines. The top and bottom sections are light blue, while the middle section is black. The nodes are represented by circles and squares of different sizes, connected by thin lines. The central text 'File I/O' is white and centered on the black background.

File I/O

Uses

- Pull in large query string arrays
- Text parsing
- Logs
- Data Export in real time

File from PHP, script, external sources





- ZSPC ASCII_File_IO (11) – read a file, read sequentially
 - `{ZSPC111: !ZSPC, ASCII_FILE_IO, READ_ASCII_line, return_code,file_name, data_loc }`
- Options
 - 1 READ_ASCII_line *Reads a line of data from an ASCII file*
 - 2 WRITE_ASCII_line *Writes a line of data to an ASCII file (!sys also)*
 - 3 READ_TEXT_answer *Reads a line of data from an ASCII file into a TEX question*
 - 4 WRITE_TEXT_answer *Writes a line of data from a TEX question to an ASCII file*
 - 5 Deprecated
 - 6 READ_SEQUENTIAL_ascii_line *Reads a line of data sequentially from an ASCII file*



Example:

Basic qpx construction

```
{ RETURN_CODE: .1 !VARIABLE, USE_PREVIOUS_answer, , }
```

```
{ FILE_NAME: .100
```

```
xxxxxx
```

```
!SPECIAL, SAVE_TEXT_line_TO_DATA, }
```

```
{ DATA_LOC: .50 !VARIABLE, USE_PREVIOUS_answer, , }
```

```
{ ZSPC11_1: !ZSPC, ASCII_FILE_IO, READ_ASCII_line, return_code,file_name,  
data_loc }
```

```
{ ZSPC11_6: !ZSPC, ASCII_FILE_IO, READ_SEQUENTIAL_ascii_line,  
return_code,file_name, data_loc }
```

Text Scrubbing Example

File
10



Profanity and Security Scrubs



A screenshot of a video player interface. At the top left is the SURVOX logo, consisting of a 3x3 grid of colored dots (purple, blue, green) followed by the word "SURVOX" in bold black text. Below the logo is a dark blue decorative bar with abstract light blue geometric patterns. The main content area is white and contains the text "Please tell us more about your recent visit." above a large, empty rectangular text input field. At the bottom center of the white area are two circular blue buttons: a pause button on the left and a play button on the right. The bottom of the screenshot shows a dark blue decorative bar with a network diagram pattern of nodes and lines.

A background network diagram with blue lines and nodes. The nodes are represented by circles and squares of varying sizes, connected by thin lines. The overall color scheme is a gradient of blue, from light to dark.

Web Services

Ajax, APIs, wget

AJAX



AJAX



AJAX



- HTML and CSS for presentation
- The Document Object Model (DOM) for dynamic display of, and interaction with, data
- XML for the interchange of data
- The XMLHttpRequest object for asynchronous communication
- JavaScript to bring these technologies together

Source: [https://en.wikipedia.org/wiki/Ajax_\(programming\)](https://en.wikipedia.org/wiki/Ajax_(programming))



FULL TEXT RESPONSE SEARCH

This allows the respondent to start typing an answer and have a list of matching selections drop down so they can select the needed response.

Which state do you live in?

Please begin by typing in the boxes below. Once you see an option you wish to select, please click on it.

State: Code:





- Many web services provide API interfaces
- wget provides Linux access to http
- Survent integration functions
 - !sys
 - !zspc



```
{ QTEXT1: 1901.18
```

Please enter your phone number and we'll send you a message.

```
!VARIABLE, , }
```

```
{ RETURNCD: 5001.1 !VARIABLE, USE_PREVIOUS_answer, , }
```

```
{ ExecuteX:
```

```
rm -f out[1-3];
```

```
wget -O out1 -o out3 'https://api.data247.com/v/2.0?
```

```
out=json&user=username&pass=acctpass&api=T&p1=:qtext1:' ;
```

```
grep "sms_address" out1 |
```

```
tr -s " " |
```

```
cut -d'"' -f4
```

```
> out2
```

```
!SYSTEM, RUN_PROGRAM }
```



```
{ FILENAME: 5011.70
out2
!SPECIAL, SAVE_TEXT_line_TO_DATA, }

{ SMSADD: 5200.100
!VARIABLE, USE_PREVIOUS_answer, , }

{ Z111: !ZSPC, ASCII_FILE_IO, READ_ASCII_line, returncd,filename,smsadd }

{ SHOWINFO:
sms address: \ | smsadd |
!DISPLAY, WAIT }

{ SENDSMS:
echo "Please visit our website at www.survoxinc.com" | mutt -F mutt.rc -- \ | smsadd |
!SYSTEM, RUN_PROGRAM }
```

API




WGET Demo

sbdev.cfmc.com/sb01/josh/wget/index.php

SURVOX

Please enter your password.



© Copyright 2015 - CFMC

Motorola Connect

MESSAGING RECENT CALLS

Auto Test
2:53 PM

Type message

retrieved 2 minutes ago

A background image featuring a network diagram with various nodes (circles and squares) connected by lines, set against a blue gradient. The diagram is centered on a black horizontal band.

External Components

DBR & DB Files

Dynamic Response Lists

DBR

- Disc Base Recode table
- Allows for very large response lists
- Has drill-down / hierarchy feature
- External to QPX/QFF, independently updatable

```
EX:
~PREPARE DBR
OUTPUT=COMP1.DBR
  01 IBM
001 (SKIPTO QQ4) A 360
      B 370 \BTESTING\E
002 X- 1
003 Y- 2
004 Z- 3
005 C 1130
      02 HEWLETT PACKARD
          AA 3000
006 1 II/III QQ5
007 2 3X
008 3 4X
      4-6X
009          A 64
010          B 68
          BB 900 Series
011 1 925
012 2 935
013 3 Other
014 03 PERTEC
      04-MISC
015          AAA MODEL 1
016          BBB MODEL 2 QQ6
???
```

Unmatched code in data

```
END
~END
```

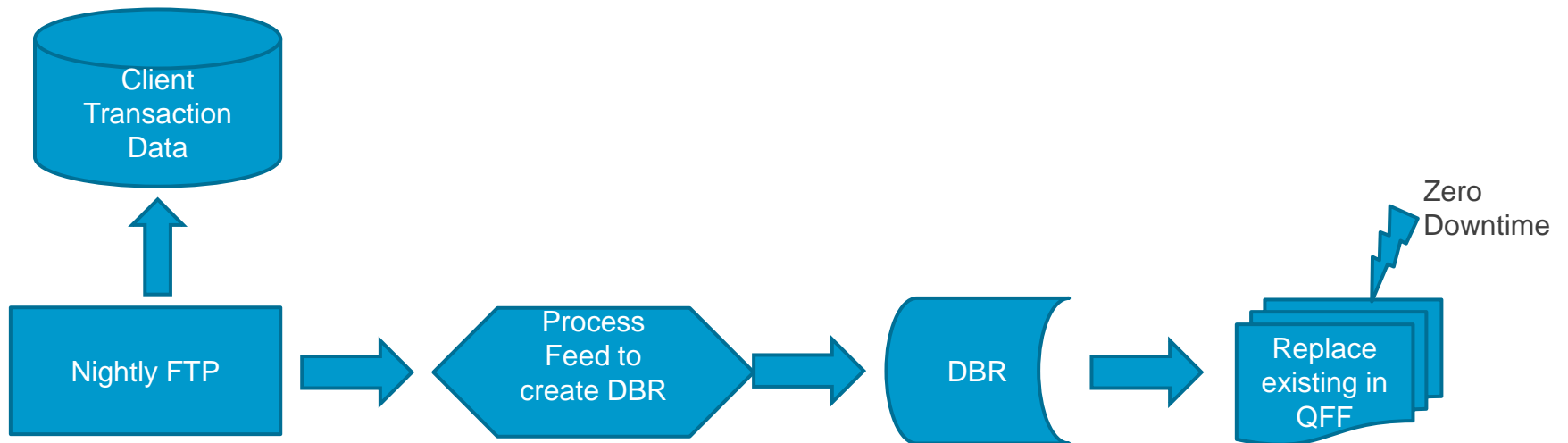
Dynamic Response Lists

DBR

```
~PREPARE DBR
OUTPUT=state_name.DBR,ENTRIES=20000
101 101    CO
102 102    MN
103 103    IL
104 104    IL
105 105    IL
106 106    IL
107 107    MN
108 108    CO
??? ??? missing
END
```

Dynamic Response Lists

DBR



- ZSPC DB_file_IO (3)– read and write a Survox db file
 - {ZSPC3: !ZSPC, DB_file_IO, WRITE_BINARY_DB_item, Rescode, zipcode, Zipdata.62 , Zdb}
- Options
 - 1 Write_Binary_DB_Item stores ASCII data from a file into a Survox DB file
 - 2 Store_ASCII_DB_Item stores binary data from a data file into a Survox DB file
 - 3 Write_ASCII_DB_Item takes an ASCII DB item and writes it to a new data file
 - 4 Write_Binary_DB_Item takes a binary DB item and writes it to a new data file

A background network diagram with blue lines and nodes on a light blue background. The nodes are represented by circles and squares of varying sizes, connected by thin lines. The diagram is centered on a black horizontal band.

OS Tools

mail, sed, bc, ?

- !sys,run_program (!sys,2)
- Executes as if at Command Line
- Simple commands or Complex Shell scripts

When the SYS,2 executes, Survent is exited, and saves the existing data case as “cfmd####” where “####” is the station number of the device executing the SYS command.

Once the external program has finished, the file cfmd#### is read back in to the memory.

EX:

```
{  
cp /home/cfmc/myfile.txt /home/cfmc/myfile2.txt  
!SYS,2 }
```



- Email with mail, mutt, sendmail, etc.
 - Hot Alerts
 - Email Invitations
 - Error Notifications
- Combine with ZSPC options
 - wget
 - Text transformation, profanity scrub
 - Complex Math
- Shell Scripts
 - Hide full quotas
 - Send out notifications
- Survent as form generator



☐ Email

- ☐ Hot Alerts
- ☐ Notification
 - Programmers
 - Supervisors
 - Project Staff

☐ Complex Math

☐ Text Message

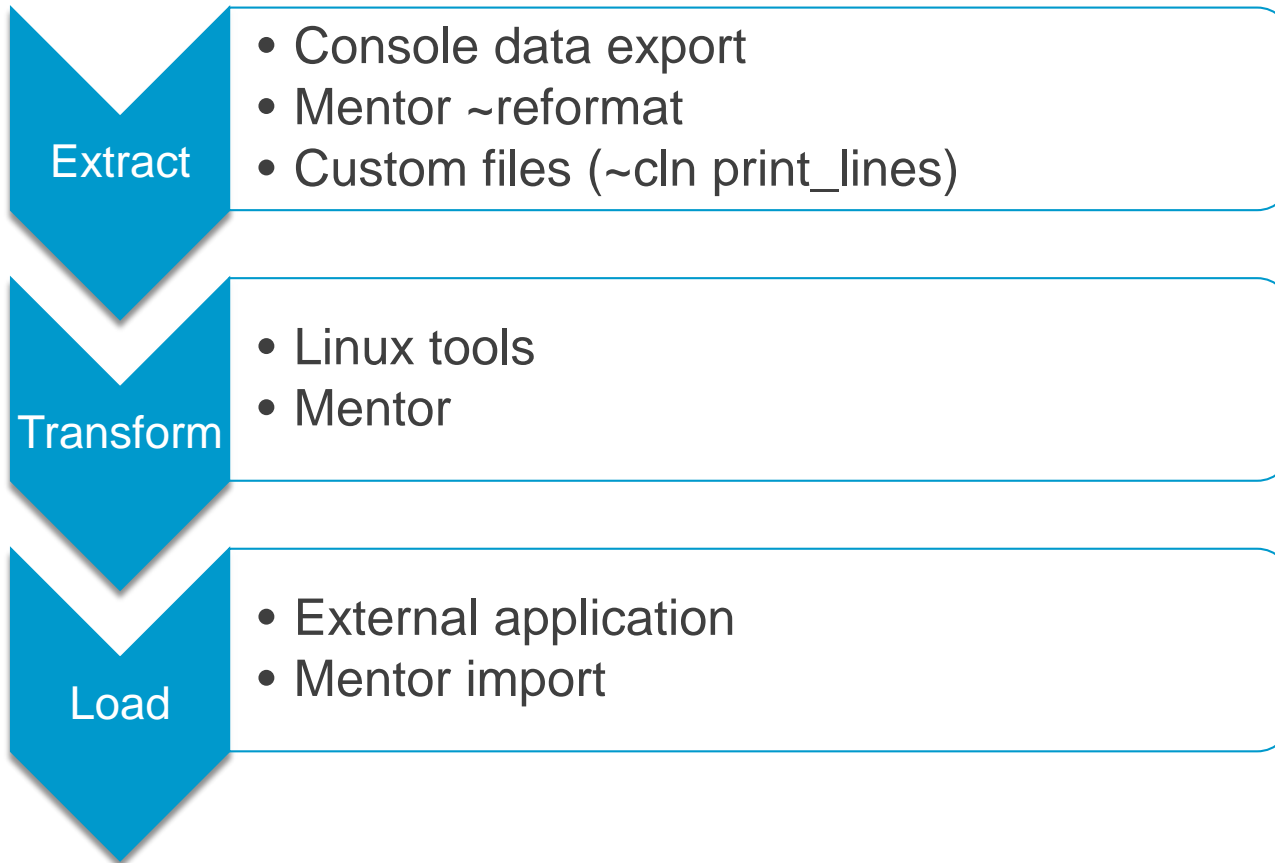
☐ PBX and Softphone integrations

A background network diagram with various nodes (circles and squares) connected by lines, set against a blue gradient. The nodes are scattered across the top and bottom sections of the slide, with a central black band containing the text.

Data Tools

Integrations and Articulation

Data Import / Export





- Tools and systems to produce quality deliverables

- ASCII
- SAS
- Delimited (CSV)
- Custom



- SPSS
- Triple-S (XML)
- Excel
- And more

- Data cleaning and consistency checking
- Crosstab data IS summarization
- Statistical testing/weighting of data (e.g., significance testing, t-testing)

Data Articulation

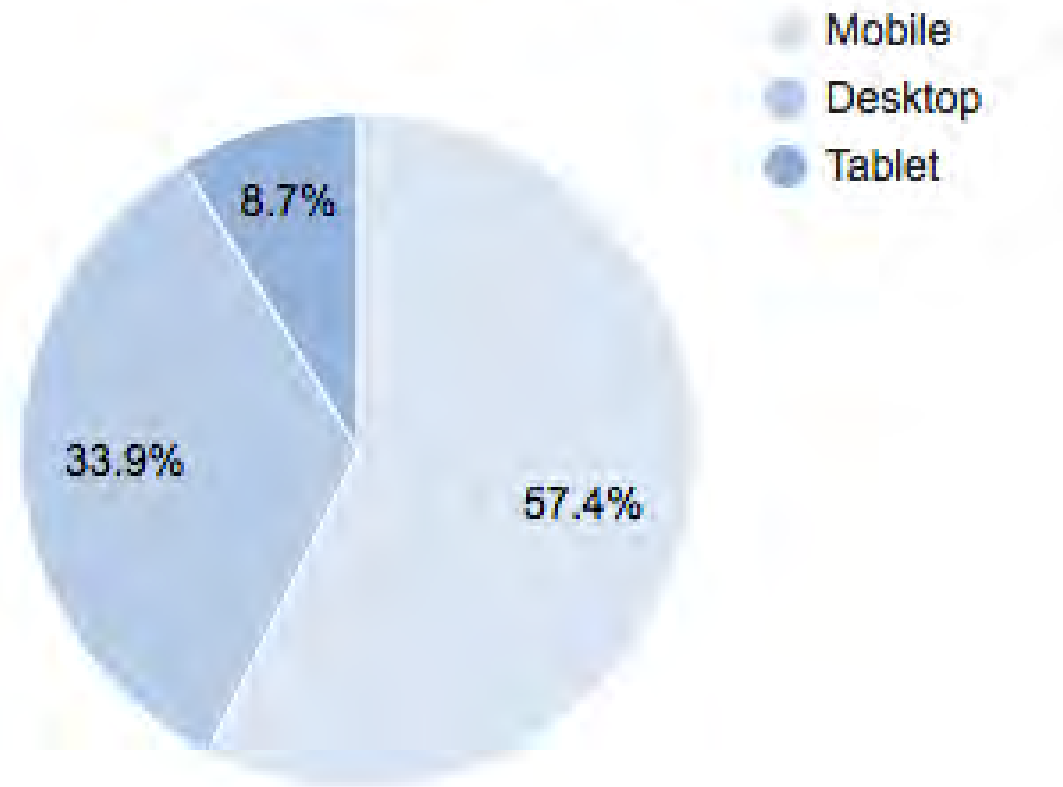


TABLE tabDev

	TOTAL
Total	920305
	100.0%
N/A	17129
	1.9%
Desktop	305805
	33.2%
Mobile	518409
	56.3%
Tablet	78962
	8.6%



June 2015 Device Analysis: Completes by Device



The background of the slide is a light blue color with a network diagram pattern. The diagram consists of numerous thin, dark blue lines connecting various nodes. Some nodes are represented by small circles, while others are small squares. The connections are dense and crisscrossing, creating a complex web-like structure. The nodes are scattered across the entire background, with some clusters and some isolated points.

ODBC



- Survent
- Mentor
- PHP
- Linux

Customer Examples

ODBC

Quotas	Use a database as a quota counter file for cross-system quota controls
Sample	Create an open web study then connect to a database to get sample information and build the fon record on the fly.
Data	Automatically update data elements from within Survent
Mentor	Run counts on tr file and database to validate ETL

Q & A



Next Steps

- Please share this information with your team
- Think about
 - How you can create a better experience for your interviewers and respondents.
 - How can you expand your business opportunities?
- Learn more at docs.Survoxinc.com
- Visit [Survox Services Center](#)
 - Contact Professional Services for custom integrations
 - Request training