Contents

Preface xi
Acknowledgments xiii

Part 1 An Introduction to SAS/IntrNet Software

Chapter 1 Overview of SAS/IntrNet and Related Technologies 3

1.1 Is the Application Dispatcher a Good Fit? 4

1.2 Components of SAS/IntrNet Software 5

1.2.1 The Application Dispatcher 5

1.2.2 SAS Design-Time Controls 5

1.2.3 Xplore Sample Web Application 6

1.2.4 htmSQL 6

1.2.5 SAS/CONNECT Driver for Java 6

1.2.6 SAS/SHARE Driver for JDBC 6

1.2.7 Tunnel Feature 7

1.3 Other SAS Web Components and Technologies 7

1.3.1 The Output Delivery System 7

1.3.2 The Web Publishing Tools and Related Macro Tools 8

1.3.3 SAS AppDev Studio, a SAS Applications Development Environment 9

1.3.4 SAS®9 Business Intelligence Platform 9

1.4 Industry Components 12

1.4.1 Scripting Languages 12

1.4.2 Dynamic HTML 15

1.4.3 Web Services 16

1.5 Component-Based Architectures 16

1.6 Terminology 18
Part 2  How the SAS/IntrNet Application Dispatcher Works

Chapter 2  Overview of the Application Dispatcher Process Flow  23
  2.1 Introduction  23
  2.2 Application Broker—Application Server Process Flow  24
  2.3 Process Flow Including the Load Manager  25
  2.4 Performance Benefits of Using the Load Manager  27

Chapter 3  The Application Broker and the Load Manager  31
  3.1 Introduction  31
  3.2 Specifying Additional HTML Output  33
  3.3 Defining the Load Manager  34
    3.3.1 The Load Manager Command  34
  3.4 Defining the SAS Application Servers  35
    3.4.1 Socket Servers  35
    3.4.2 Pool Servers  37
    3.4.3 Launch Servers  39
  3.5 Application Broker Process Flow  40
    3.5.1 Load Manager Administrative Functions  46

Chapter 4  The Application Server  47
  4.1 Introduction  47
  4.2 SAS Application Server Executives  49
  4.3 Application Server Sessions  50
    4.3.1 ODS and Sessions  50
  4.4 PROC APPSRV  51
    4.4.1 TCP/IP Port  51
    4.4.2 Assigning Libraries  51
    4.4.3 Initiating and Terminating Requests  54
  4.5 Application Server Process Flow  55
    4.5.1 Application Server Functions Available to the Executing Program  60
    4.5.2 Application Server Clean-Up Processing  60
Chapter 5 Communicating with the Application Dispatcher 63

5.1 Introduction 63
5.2 Name/Value Pairs Defined by the User’s Request 64
   5.2.1 _program: The SAS Program to Execute 64
   5.2.2 _service: The Application Server to Process the Request 65
   5.2.3 _debug: The Output to Display 65
   5.2.4 Program-Specific HTML Name/Value Pairs 68
5.3 Name/Value Pairs Defined by the Application Broker 72
   5.3.1 Application Broker Administrative Fields 72
   5.3.2 Environment Variables 73
   5.3.3 Installation Dependent Fields 74
5.4 Name/Value Pairs Defined by the Application Server 76

Part 3 Developing Application Dispatcher Programs

Chapter 6 Methods You Can Use to Access and Reference Input Parameters 81

6.1 Introduction 81
6.2 Accessing Parameters as Macro Variables 82
   6.2.1 Using and Referencing the Multiple Values for a Single Parameter (the Suffix Variables) 85
   6.2.2 Using the APPSRV_UNSAFE Function 87
6.3 Accessing Parameters in SCL Programs 91

Chapter 7 Various Techniques to Generate HTML 94

7.1 Introduction 94
   7.1.1 The Problem with Generating Valid HTML 95
7.2 Simple PUT and FILE Statements 96
   7.2.1 The generateFormTag and generateInputTag Sample Macros 101
7.3 Extending the Output Delivery System 102
   7.3.1 FORM Tags Via TITLE and FOOTNOTE Statements 102
   7.3.2 Character Variables with HTML Form Text 105
7.4 SCL Submit Blocks 107
7.5 Including Static HTML from External Sources 110
   7.5.1 A Macro Tool to Include External HTML 114
7.6 SAS Server Pages  116
  7.6.1 The SAS Server Page Macro  116
  7.6.2 Sample Server Page: List Libraries  118
  7.6.3 A Macro to Generate Data-Driven SELECT Tags  119
  7.6.4 Sample Server Page: List the Data Sets in the Selected Library  121
  7.6.5 Sample Server Page: List the Variables in the Selected Data Set  122
  7.6.6 A Macro to Generate Checkboxes for Variables in a SAS Data Set  124
  7.6.7 A Program to Page Through the Selected Data Set  125

7.7 SAS Design-Time Controls  127

Chapter 8 Creating Pages with Mixed and Alternative Content Types  131

8.1 Introduction  131
8.2 Defining the Content Type to Be Generated  132
  8.2.1 Other Headers: Selected Examples  132
8.3 Generating Pages with Other Content Types  135
  8.3.1 Creating a Comma-Separated Values File  136
  8.3.2 Downloading Reports and Results into Microsoft Excel  137
  8.3.3 Generating Content for Printing  138
  8.3.4 Generating Multiple Output Types at Once  143
8.4 Generating Pages with Mixed Content Types: Text and Graphics  146
  8.4.1 Ensuring That the Graph Is Current  153

Chapter 9 Using REQUEST INIT and REQUEST TERM to Specify Set-up and Shut-down Behavior  155

9.1 Introduction  155
9.2 Specifying the REQUEST INIT and REQUEST TERM Programs  156
9.3 Using REQUEST INIT and REQUEST TERM  156
  9.3.1 Defining Libraries and Files  157
  9.3.2 Overriding or Supplying Parameter Values  159
  9.3.3 Standard Header and Trailer Blocks  161
  9.3.4 Terminating a Request  163
Chapter 10  How to Create and Use Sessions  165
  10.1  Introduction  165
  10.2  Creating a Session  166
  10.3  Saving or Restoring All Macro Variables  171
  10.4  Returning to a Previous State of a Session  175
  10.5  Generating Friendlier Messages for Expired Sessions  179
  10.6  Extending Sessions  182
  10.7  Session INIT and TERM Programs  186

Part 4  Addressing Common Application Requirements

Chapter 11  Tools and Techniques for Debugging  191
  11.1  Introduction  191
  11.2  Using the _debug Parameter  192
  11.3  The PROC APPSRV LOG Statement  196
  11.4  Conditionally Generating Debugging Output  197
  11.5  Running Application Dispatcher Programs Using SAS Display Manager  200
        11.5.1  Special Handling for SCL Programs  201
  11.6  Dedicating an Application Server for Debugging  202

Chapter 12  Tips for Safeguarding Security  205
  12.1  Introduction  205
  12.2  The Application Server Environment  206
        12.2.1  Limiting Which Application Brokers Can Access an Application Server  206
        12.2.2  Hiding Passwords  208
        12.2.3  Best Practices for a Secure Application Server Environment  211
  12.3  Controlling Access to Data and Reports  212
        12.3.1  Customizing Menu Choices  219
        12.3.2  Using AUTH=HOST  223
Chapter 17  Techniques for Handling Long-Running Processes  267

17.1 Introduction  267
17.2 Using the Cascading Style Sheets Display Attribute 268
17.3 Using the JavaScript location.replace Function  271

Chapter 18  Using Scheduled Execution to Handle Long-Running Processes  275

18.1 Introduction  275
18.2 Sample Implementation  276
18.3 Doing More with Scheduled Execution  279

Chapter 19  Handling On-Demand Long-Running Requests  281

19.1 Introduction  281
19.2 E-mailing Results from an On-Demand Long-Running Process  282
  19.1.1 Notifying the User  284
19.2 Updating Process Status by Refreshing the User’s Browser 285
  19.2.1 Notifying the User  287
19.3 The Sample Framework—Spawning a Separate SAS Session  290
  19.3.1 Sample Program to Submit a Long-Running Program—spawnSAS.source  295
  19.3.2 Sample spawnSAS Macro  298
  19.3.3 Sample Long-Running Program  300
19.4 Doing More with Independent Sessions  305

Chapter 20  Using Metadata-Driven Reporting to Construct Reports  307

20.1 Introduction  307
20.2 Sample Implementation  309
  20.2.1 The Logon Template  311
  20.2.2 The Logon Program  312
  20.2.3 Selecting the Case or Individual to Examine  315
  20.2.4 Report Components  319
  20.2.5 The assembleReport Macro  323
Chapter 21  Packaging the Application Dispatcher as a Web Service  327
  21.1 Introduction  327
  21.2 Solution Architecture  329
  21.3 Sample Application  330
    21.3.1 The Sample .NET Client Application  331
    21.3.2 The .NET Web Service  332
  21.4 Using the Sample Client Application  335
  21.5 Final Thoughts  338

Index  339