

レイクサイドソフトウェア：

ホワイトペーパー

# SysTrack を使用したソフトウェア 資産の分析

## 目次

はじめに .....	3
各ユーザーはどんなソフトウェアを実際に使っているか? .....	3
各ユーザーが使用しているソフトウェアのパフォーマンスは? .....	5
各ユーザーのソフトウェアはどんな機能を果たす必要があるか? .....	6
結論 .....	8
詳細情報 .....	8

## はじめに

ワークスペース分析の重要な構成要素の 1 つとして、当該環境におけるソフトウェア資産を監視して把握することがあげられます。エンドユーザーが日常業務で使用するソフトウェアとの対話をどのように行っているかを把握し分析することが、ソフトウェア資産管理の中核となります。ソフトウェア資産の分析対象は大まかに、ソフトウェアの利用、パフォーマンス、依存性という 3 つのカテゴリに分けられます。「利用」というカテゴリは、当該環境におけるライセンスの最適化やアプリケーション配信の計画立案をカバーします。「パフォーマンス」というカテゴリは、リソースの消費、および最適な運営を実現するための全体的な要件に関係します。「依存性」というカテゴリは、意図した通りに機能するようソフトウェアを維持するためには、どんなモジュールやバックエンド支援コンポーネントが必要となるかに関する洞察を提供します。これらのカテゴリはそれぞれ、健全で、適切に管理され、最終的に成功を収めるエンドユーザー環境を維持するための役割を演じます。

## 各ユーザーはどんなソフトウェアを実際に使っているか？

ソフトウェアの利用状況を監視できることは、企業にとって極めて重要です。この能力は、ソフトウェアの配布とライセンスの最適化を中心として、ローカルに配信すべきアプリケーションの決定から、環境における混乱の一掃、および特定の職務上の役割に対してソフトウェアライセンスを配布する最適な方法の決定に至るまでのあらゆる物事に関係します。SysTrack の Software Asset Analytics Kit として提供される複数のダッシュボードを使用すると、管理者は、アクセスされたアプリケーションがどんな場合に、どれだけの頻度で使用されるかについての観測された客観的データに基づいて、自社のエンドユーザーを最適にプロビジョニングする方法を明確に把握できるようになります。これにより、管理者は、プロビジョニング、ライセンスの最適化、ソフトウェアパッケージの確実な配信に関するさまざまな意思決定を簡単に行えるようになります。この結果、最終的に、利用頻度の低いアプリケーションを除去することにより、ライセンス費用を引き下げることができます。また、すべてのユーザーがほぼ同一のソフトウェアポートフォリオを必要としており、しかもそのポートフォリオが金銭面で大幅に有利な別のライセンスによってより効果的にカバーされることに管理者が気付いた場合には、ライセンスングを合理化できます。

また、個々のユーザーやシステムによるソフトウェア使用状況に関する洞察を得ることは、配信計画とライセンスングに基づいて職務上の役割を最適化する場合にも役立ちます。比較的汎用性のあるパッケージのセットをほとんどの環境に配信する代わりに、IT 管理者は、個々の職務上の役割に最適となるように配信されるソフトウェアをカスタマイズするツールを利用できます。最適化されたプロビジョニングにより、例えば互換性が理由でマイナスの影響を持つ可能性のある過剰なソフトウェアを整理するなどして、さまざまな方法でより成功を収めるような環境を構築できます。さらに、当該環境のソフトウェアポートフォリオを削減し同ポートフォリオをより適切に保守するならば、問題が発生する可能性のある領域の数を制限することにより、テクニカルサポートやヘルプデスクのスタッフにかかる負担を軽減し、最終的に時間とコストを節約できるようになります。

例えば、当社の IT 管理者であるジーン (Gene) は、当社の環境でソフトウェアのプロビジョニングを最適化するにはどうすればよいのかをずっと考えてきました。彼は SysTrack の要約利用状況ダッシュボードを使い始めたところ、インストールされている上位 50 個のパッケージのうち、頻繁に使われているものはほんのわずかであることに気付きました。そこで、まずこれらの使われていないパッケージを減らすことから始めることに決めました。続いて同ダッシュボードをドリルダウンしてより詳細なデータにたどり着くと、インストールされているシステムのうち 50%未満しか使われていないパッケージだけがソフトウェア詳細パネルに表示されるように設定しました (図 1)。

Software Details - Select Software to view Installed Systems Details

Show All/Used/Unused Software Packages: All

Show Software with Use Percentage between: 0 50

Software Search:

Software	Publisher	Version Count	Associated Executables	Used System Count	Installed Systems	Use Percentage	Use Status
Microsoft Silverlight	Microsoft Corporation	4	2	4	28	14.29	Used
@(Windows.PrintDialog_6.2.0.0_neutral_neut		1	1	8	27	29.63	Used
OneNote		1	1	1	23	4.34	Used
Cisco WebEx Meetings	Cisco WebEx LLC	1	3	7	14	50	Used
Mozilla Maintenance Service	Mozilla	6	1	1	12	8.33	Used
Microsoft SQL Server 2014 Setup (English)	Microsoft Corporation	5	1	1	12	8.33	Used
Microsoft Visual Studio 2010 Shell (Isolated) -	Microsoft Corporation	1	2	3	12	25	Used
7-Zip 16.02 (x64)	Igor Pavlov	1	2	3	8	37.5	Used

図 1 : 利用率が 50%未満のソフトウェア

利用率の低いアプリケーションをメモすることで、ジーンはすでに、現在の環境がおそらく過剰にプロビジョニングされていることに気が始めました。続いて、注目する対象となるソフトウェアパッケージを選択することにより、この調査を絞り込みました。次の 2 つの表には、選択されたパッケージの各バージョンとそれがインストールされているシステムの数（図 2）と、選択されたバージョンのパッケージが各システムでどのように利用されているかの詳細（図 3）が示されています。

Selected Software Versions

Version	System Count
5.1.20513.0	1
5.1.41212.0	1
5.1.50901.0	9
5.1.50905.0	17

図 2 : 選択されたバージョンのソフトウェアがインストールされているシステムの数

SystemName	Primary User	Asset Status	Last Scan	Install Date	Install Size (MB)	Last Used Date
EHEA8LAKESIDESOFTWARE.ORG		Deployed	2017-03-22	2017-03-16	299.12	
URLSFILE8LAKESIDESOFTWARE.ORG	ISS APPROOLV	Deployed	2017-03-22	2017-03-16	448.11	
EHEA8CTLAKESIDESOFTWARE.ORG	LSJalnel	Deployed	2017-03-22	2017-03-15	795.64	
URLSDESK5LAKESIDESOFTWARE.ORG	LSJandrewh	Deployed	2017-03-22	2017-03-15	101.47	2017-03-20 09:22:29
URLSLAP11LAKESIDESOFTWARE.ORG	LSJlawnd	Deployed	2017-03-22	2017-03-14	448.11	
URLSDCS1LAKESIDESOFTWARE.ORG	LSJgawng	Deployed	2017-03-22	2017-03-16	398.45	
URLSDESK1LAKESIDESOFTWARE.ORG	LSJgawng	Deployed	2017-03-22	2017-03-15	101.46	
URLSDESK3LAKESIDESOFTWARE.ORG	LSJgawng	Deployed	2017-03-22	2017-03-14	101.47	
URLSDESK4LAKESIDESOFTWARE.ORG	LSJgawng	Deployed	2017-03-22	2017-03-16	101.47	
URLSLAP21LAKESIDESOFTWARE.ORG	LSJmell	Deployed	2017-03-22	2017-03-20	299.12	2017-03-22 06:55:49

図 3 : 選択されたアプリケーションの各システムにおける利用状況

これらの結果から、ジーンは、利用されていないパッケージを保持しているシステムとユーザーを把握できました。続いて、システムベースのダッシュボードを通じて、各システムの利用状況を確認できました。ソフトウェアパッケージのうちどれが使われておらず、どれが最も使われているかに関するより詳しい情報を取得し、考慮すべき必要のある技術的な側面を理解することで、ジーンは、これらのシステムやユーザー向けにプロビジョニングを最適化するための知識を得ることができました。これにより、システムが過剰にプロビジョニングされることは無くなり、その結果、環境が大幅にクリーンになり、以前よりも管理しやすくなりました。

## 各ユーザーが使用しているソフトウェアのパフォーマンスは？

ソフトウェアのパフォーマンスは複雑なトピックであり、現在の環境やアプリケーションの配信手法がユーザーのニーズを十分に満たしているかどうかを判定するために役立つ各種の主要な数値指標の利用に關与するものです。パフォーマンスは、エンドユーザーがまさにアプリケーションを利用する場所である対話ポイントに最も近い属性であるため、ソフトウェア資産分析にとって最も重要な側面の 1 つとなります。概念的に言えば、このカテゴリーには、「このアプリケーションがクラッシュし続ける理由は？」や「このアプリケーションのローディングには長い時間がかかっているか？」などのような、パフォーマンスに関する多くの一般的な疑問に対する回答を探ることが含まれます。これらの疑問に答えるためには、リソース消費量の詳細（CPU、メモリ、IOPS、ネットワーク帯域幅など）、障害の特性、起動にかかった時間、およびその他のインジケータを含む、ソフトウェアに関連付けられている主要なパフォーマンス数値指標を理解することが必要となります。

当社の IT 管理者であるジーンが、あるアプリケーションをアップデートした後、同アプリケーションが当該環境でどのように対話を行うかを確認したいとします。彼はダッシュボードを使用して、当該環境にある選択されたアプリケーションの要約情報を表示します。続いてそのダッシュボードをドリルダウンすることでバージョンごとの詳細情報を見つけ、新しいバージョンのメモリ消費量が古いバージョンよりも高いことに気付くことができます（図 4）。

Display Name	Application Version	Execution Count	AvgActiveCPUMIPS	AvgTotalCPUMIPS	AvgActiveMemMB	AvgTotalMemMB	AvgIOPS	AvgIOKbPerSec	FaultCount	ErrorCount	HangCount
Google Chrome (chrome.exe)	54.0.2840.99	2273	1594.53	323.31	240	122.48	0.56	0.41	0	0	0
Google Chrome (chrome.exe)	55.0.2883.87	37761	842.75	34.04	251.22	78.97	0.2	0.21	0	0	0
Google Chrome (chrome.exe)	56.0.2924.87	89434	1101.58	49.13	228.3	81.88	0.14	0.15	4	0	4

図 4 : バージョンごとのアプリケーションのパフォーマンスの詳細

さらに同ダッシュボードを通じて、ジーンは、各システムにおける特定のアプリケーションのバージョンの CPU、メモリ、IO に関するリソース消費量を表示することで、当該バージョンが個々のシステムと現在どのように対話を行っているかを確認できます。例えば、図 5 に示されている表から、2~3 のシステムのメモリ消費量が異常に多いことに気がきます。

SystemId	ExecutionCount	AvgActiveCPUMIPS	AvgTotalCPUMIPS	AvgActiveMemMB	AvgTotalMemMB	AvgIOPS
APP-WIN10-RUCH.LAKESIDESOFTWARE.ORG	2761	1424.6	265.18	220.48	93.3	0
BENVIN10PC.LAKESIDESOFTWARE.ORG	19417	630.04	17.47	244.65	91.95	0
GEOFF-SURFACEPRO4.LAKESIDESOFTWARE.ORG	3620	856.87	24	118.78	33.96	0
JAN2012.LAKESIDESOFTWARE.ORG	110	424.08	2.26	86.32	37.57	0
LSISE-01.LAKESIDESOFTWARE.ORG	572	450.21	22.86	411.77	155.13	0
LSIXA76PVS-P1.LAKESIDESOFTWARE.ORG	1281	442.82	22.84	353.97	137.92	0
LSIXA76PVS-P2.LAKESIDESOFTWARE.ORG	1364	526.27	27.3	395.34	137.33	0
LSIXA76PVS-P11.LAKESIDESOFTWARE.ORG	360	253.36	15.9	136.78	62.5	0
LSIXA7-P1.LAKESIDESOFTWARE.ORG	2563	734.46	40.41	296.17	81.88	0

図 5 : 各システムにおける特定バージョンのアプリケーションの利用状況に関する詳細

この発見の後、ジーンはユーザーおよびユーザーのリソース消費量の詳細へとドリルダウンすることで、リソース消費量上昇の原因となっているユーザーを特定できます。要約データから始まって最終的にパフォーマンスへの影響を引き起こした原因にたどり着くことができたため、ジーンはその問題に対処した後、環境を監視することでパフォーマンスに対する影響をさらに詳しく調査できるようになります。

このような数値指標を観測することは、環境における問題の原因を究明するのに役立つだけでなく、ソフトウェアアップデートやシステム更新などの変化がマイナスの影響を持たないことを保証するために定常状態の監視を行うためにも役立ちます。もしそれらがマイナスの影響を持つ場合、その問題に対して迅速に対処し、今後その問題を回避できます。ほとんどのアップデートはエンドユーザーに対して大きな影響を与えないように見えますが、それをどうやって確認したらよいのでしょうか？影響が発覚するまで待つ代わりに、IT 管理者は、変更が発生した際にパフォーマンスに関する数値指標を観測することで、アップデートが非互換であるかどうかを判定できます。このような分析に基づいて IT 管理に関する指示を提供することで、パフォーマンスの問題を迅速かつ容易に診断し、平均的なユーザーのリソース消費量を完全に把握することが可能となります。

## 各ユーザーのソフトウェアはどんな機能を果たす必要があるか？

ソフトウェアの依存性を理解することで、「自分のソフトウェアはどんな機能を果たす必要があるか？」という一見複雑な疑問に答えることができます。要件はさまざまな場合がありますが、理解すべき中心属性はどんな場合でも、アプリケーション接続、必要なシステムコンポーネント、互換性、アプリケーション利用



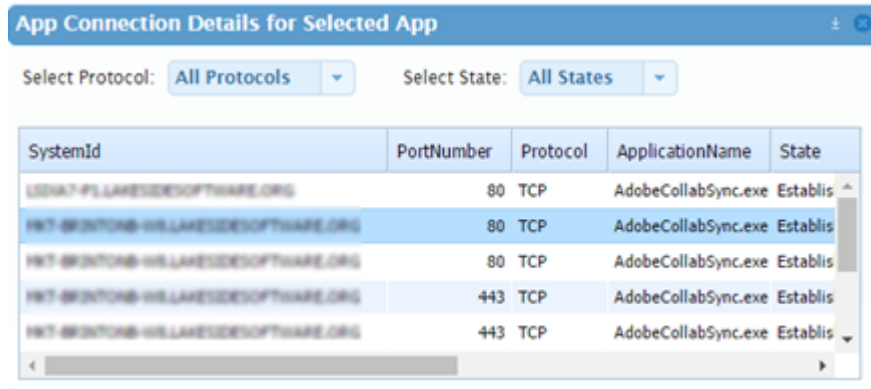
率となります。場合によっては利用不可のポートやソフトウェアパッケージ間における非互換のシステムコンポーネントのような各種の制限が存在することがあるため、これらのコンセプトは、健全で成功を収める環境を達成するために重要となります。例えば、ネットワーク接続を行うアプリケーションの中には、実行するために特定のポートとプロトコルを必要とするものがありますが、セキュリティ上の理由から、または他のアプリケーションが必要とする接続でポートが占有されていることが理由で、それらのポートやプロトコルを利用できない場合があります。

IT 管理者であるジーンが「Software Dependencies Summary」ダッシュボードを使って、トップレベルから分析を開始したとします。多くのシステムにインストールされているパッケージが実際にはほとんど利用されていない場合もあることに気づき、それらを自分のターゲットとすることを決めました（図 6 を参照）。



図 6 : アプリケーションをインストールしているシステムの数と実際にそのアプリケーションを使用しているシステムの数比較

続いてジーンは同ダッシュボードを使って、十分に利用されていないだけでなく、接続の開始に関して潜在的な問題となる依存性をサポートすることが困難であるソフトウェアパッケージを特定しました。この結果、各ソフトウェアパッケージに関する詳細情報（同パッケージが最後に利用された日時や、それが保持するネットワーク接続数など）を確認できるようになりました。また、接続を必要とするアプリケーションを含んでいる最も利用されているソフトウェアパッケージが、制限されているポートやプロトコルを使用していないことを確認できました（図 7 を参照）。



The screenshot shows a window titled "App Connection Details for Selected App". It has two dropdown menus: "Select Protocol: All Protocols" and "Select State: All States". Below is a table with the following data:

SystemId	PortNumber	Protocol	ApplicationName	State
LDNA7-PL.LAKESIDESOFTWARE.ORG	80	TCP	AdobeCollabSync.exe	Establis
NET-8RINTONB-W.LAKESIDESOFTWARE.ORG	80	TCP	AdobeCollabSync.exe	Establis
NET-8RINTONB-W.LAKESIDESOFTWARE.ORG	80	TCP	AdobeCollabSync.exe	Establis
NET-8RINTONB-W.LAKESIDESOFTWARE.ORG	443	TCP	AdobeCollabSync.exe	Establis
NET-8RINTONB-W.LAKESIDESOFTWARE.ORG	443	TCP	AdobeCollabSync.exe	Establis

図 7 : アプリケーション接続状況の確認

必要なシステムコンポーネントに関する調査をさらにもう少し行うことで、ジーンは、当該環境におけるソフトウェアパッケージがどのように機能する必要があるかに関してほぼ完全に理解できるようになりました。そして最終的に、彼はこの知識に基づいて、十分に利用されていないアプリケーションや非互換のアプリケーションを削除し、制限を回避し干渉を減らすようにネットワーク接続を構成することで、ソフトウェアパッケージの提供をさらに最適化できるようになりました。

## 結論

資産の最適化はコスト削減をもたらすため、ソフトウェア分析はワークスペース分析において非常に重要な位置を占めています。ソフトウェア資産の分析は複雑なトピックですが、その分析対象は大まかに、ソフトウェアの利用、パフォーマンス、依存性という 3 つのカテゴリーに分けられます。SysTrack Software Asset Analytics Kit は、このような分野に関する高レベルの要約と監視をはじめ、詳細な調査を提供する優れたツールです。ソフトウェアのパフォーマンスの状態、ソフトウェアの利用状況、およびソフトウェアの依存性を理解することで、より成功を収める環境を構築できるほか、エンドユーザーエクスペリエンスを向上させることができます。

## 詳細情報

SysTrack とレイクサイドソフトウェアの詳細については、<http://www.lakesidesoftware.com/jp> をご覧ください。



レイクサイド ソフトウェア株式会社

[www.lakesidesoftware.com/jp](http://www.lakesidesoftware.com/jp)

〒100-6208

東京都千代田区丸の内 1-11-1 パシフィックセンチュリープレイス丸の内 8F

© Lakeside Software, Inc. 1997-2017. Lakeside Software および SysTrack は、Lakeside Software の登録商標または商標です。その他の社名、商品名はそれぞれの所有者の商標または登録商標です。

