Auction Design for Indirect Contentions
14 November 2014

Executive Summary

This paper recommends an ascending clock auction design for resolving string contention among competing applications for new gTLDs in contention sets where indirect contentions are present. An end-of-round price is established for each round. Any application that meets the end-of-round price remains in the auction. If an application does not meet the end-of-round price and if there is another application that bid higher and is positioned the same or better than this application, then the application that does not meet the end-of-round price is eliminated from the auction. If neither of these conditions is satisfied, then an application remains in the auction if and only if it is part of a feasible assignment of remaining applications whose bid amounts sum to at least the end-of-round price. The auction concludes when at most one feasible assignment remains in the auction. The winner(s) are then determined by maximizing bid amounts over all feasible assignments of remaining applications.

1. Background

The vast majority of competing applications for new gTLDs are in "simple" contention sets: every application of the contention set is in direct contention with every other application of the contention set. These string contentions can efficiently and fairly be resolved using simple ascending clock auctions. However, as of this writing, there are four contention sets that include indirect contentions:

- GAME / GAMES;
- SPORT / SPORTS;
- WEB / WEBS; and
- SHOP / SHOPPING / 通販.

This document has been prepared by Power Auctions LLC, the auction consultant retained for the new gTLD program, in close consultation with ICANN staff, to recommend an auction design for resolving the contention sets that include indirect contentions.

2. Descriptions of the relevant graphs

Graphical images of the various contention sets can be found at the following URL:

https://gtldresult.icann.org/application-result/applicationstatus/stringcontentionstatus .

Each contention set may be treated as a graph, where each vertex corresponds to an application and where two vertices are connected by an edge if the applications are in direct contention with one another. Viewed as such, "simple" contention sets are *complete graphs*, i.e., graphs in which every pair of distinct vertices is connected by an edge.[1] An example of a complete graph with seven vertices is shown in Figure 1:

[1] See, for example, Douglas B. West, *Introduction to Graph Theory*, Prentice Hall, Second Edition (2001) or the Wikipedia entry for "Complete graph".

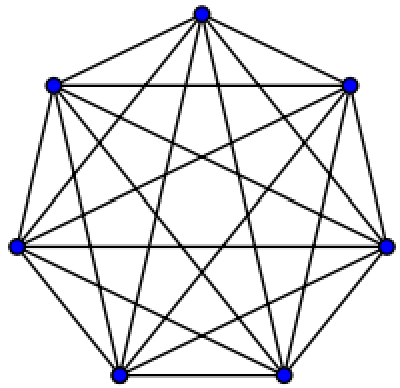

Figure 1

Using this vocabulary, as of this writing, there are currently four contention sets whose graphs are not complete.

While the problem of creating a suitable auction design for general graphs is extremely difficult, observe that the four contention sets at issue have some common simplifying features:

- There is one unique vertex (application) through which all contentions pass, in the sense that removing this vertex would result in two or more unconnected graphs. In graph theory, this is referred to as a *cut vertex*.[2]
- All of the vertices (applications) on one side of this vertex are in direct contention with one another and with the cut vertex, but with nothing else.[3]

Let us introduce some terminology that will assist with our description of auction procedures:

- If applications X and Y are in direct contention with one another—and if exactly the same other applications are in direct contention with both applications X and Y—then we will say that applications X and Y are *positioned the same* as each other.
- If applications X and Y are in direct contention with one another—and if the set of other applications that are in direct contention with application Y is a strict subset of the set of other applications that are in direct contention with application X—then we will say that application Y is *positioned better* than application X. [4]
- If application Y is positioned better than application X, we can equivalently say that application X is *positioned worse* than application Y.

[2] See, for example, West (2001) or the Wikipedia entry for "Vertex (graph theory)". In the GAME / GAMES contention set, the Charleston Road Registry Inc. application is the cut vertex. In the SPORT / SPORTS contention set, the SportAccord application is the cut vertex (but there is also a community priority). In the WEB / WEBS contention set, the Web.com Group, Inc. application is the cut vertex. In the SHOP / SHOPPING / 通販 contention set, the Commercial Connect LLC application is the cut vertex.
[3] For example, in the GAME / GAMES contention set, the other applications for .GAME (besides the Charleston Road Registry Inc. application) satisfy this condition.
[4] For example, in the SHOP / SHOPPING / 通販 contention set, all applications for .SHOP except for the Commercial Connect LLC application are positioned the same as each other—and they are each positioned better than the Commercial Connect LLC application. Similarly, the application for .SHOPPING by Uniregistry, Corp. is positioned better than the application for .SHOPPING by Sea Tigers, LLC, since the latter is in direct contention with the Commercial Connect LLC application, while the former is not.

In other words, application Y is positioned better than application X if: (a) every other application that is in direct contention with Y is also in direct contention with X; and (b) there is at least one other application in direct contention with X that is not in direct contention with Y.

Finally, we define what is meant by a feasible set of applications. A set *S* of applications is said to be *feasible* if none of the applications within *S* are in direct contention with one another. Thus, all applications within a feasible set can be awarded.

3. Ascending clock auction design: winner determination

We now describe an ascending clock auction design that can be used for the four contention sets at issue. (More generally, the design can be used for any contention set satisfying the simplifying conditions described above—and perhaps more widely.) For purposes of each round, an application either *remains in* the auction or is *eliminated from* the auction. Elimination from the auction is irrevocable in the sense that, once an application is eliminated, its status can never change back so that it remains in the auction. In each round, there is a common end-of-round price for every application remaining in the auction. After the bidding for the round ends, the remaining applications are processed as follows:

(1) Any application X that remained in the auction in the previous round and that, in the current round, met the end-of-round price, remains in the auction in the current round.
(2) Suppose that application X remained in the auction in the previous round but did not meet the end-of-round price of the current round. Further suppose that there is another application Y such that: (a) application Y is positioned the same or better than application X; and (b) the bid for application Y is greater than the bid for application X. Then application X is eliminated from the auction after the current round.
(3) Suppose that neither condition (1) nor condition (2), above, are satisfied for application X that remained in the auction in the previous round. We then ask whether application X is part of a feasible set *S* of applications that remained in the auction in the previous round (and had not already won) such that the sum of the bids of the applications in set *S* equals at least the end-of-round price of the current round. If the answer is "Yes", then application X remains in the auction after the current round; if the answer is "No", then application X is eliminated from the auction after the current round.

We then consider the entire set of applications that are determined to remain in the auction:

- If the applications that remain in the auction are together infeasible (i.e., if two or more of the applications remaining in the auction are in direct contention with one another), then the auction progresses to a next round.
- If the applications that remain in the auction are together feasible (i.e., there are no applications remaining in the auction that are in direct contention with one another), then the current round is deemed to be the final round of the auction (except for any need for tie-breaking). From the applications that remained in the auction after the previous round and had not already won, the auctioneer will determine the feasible set *S* of applications that maximizes the sum of the bids. The applications within this feasible set *S* are deemed to be the winners. Observe that any application satisfying condition (1) or (3) above in the final round will necessarily be selected as a winner. In the event that two or more conflicting sets maximize the sum of the bids, we will proceed to a tie-breaking round to resolve the tie.

4. Ascending clock auction design: pricing

In this section, we describe a pricing rule that is "second-price" in nature. This means that, to the extent that a feasible set of applications bids higher prices than was necessary for them to win, the set of applications receives a commensurate reduction in price. Second-price rules are advocated since they maximize the incentives for truthful bidding; they simplify bidding; and they tend to yield efficient outcomes.

We first state the second-price rule for the simplest possible scenario with indirect contentions: the set of applications is {A, B, C}; there is a choice between application B and applications A and C (which can coexist):

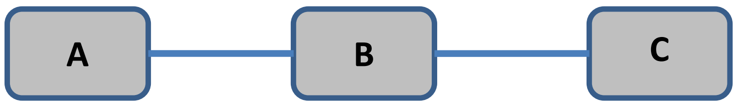

Figure 2

In the following statements of second-price rules, p_A refers to the last relevant bid by application A, p_B refers to the last relevant bid by application B, etc.

Second-price rule for the "A – B – C" scenario of Figure 2:

- If $p_B > p_A + p_C$, then application B wins. It pays the amount $p_A + p_C$. This is entirely consistent with the second-price rule we use for simple contention sets.
- If $p_A + p_C > p_B$, then applications A and C win. Together, they pay p_B . Again, this is entirely consistent with the second-price rule we use for simple contention sets. However, there is a cost allocation problem between applications A and C. We resolve this by requiring each application to pay only its proportionate share of the total amount needed to defeat application B, i.e.,
    - o Winning application A pays $\left(\frac{p_A}{p_A+p_C}\right)p_B$.
    - o Winning application C pays $\left(\frac{p_C}{p_A+p_C}\right)p_B$.

Next, we state the second-price rules for more complex scenarios. For example, this will treat the configuration depicted in Figure 3: there are four applications {A1, A2, B, C}; the "maximal" feasible sets are B by itself, applications A1 and C together, or applications A2 and C together. These rules will also apply to other complex scenarios with indirect contentions.

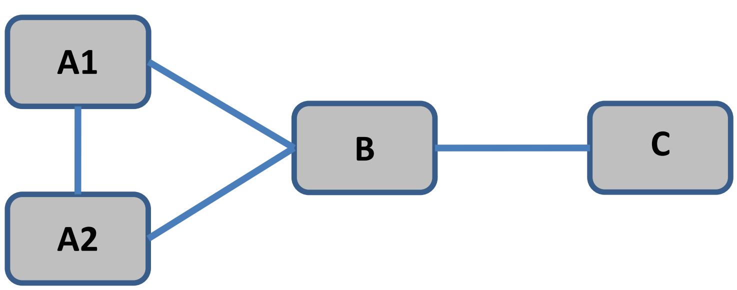

Figure 3

Second-price rule for complex scenarios:

- If winning application A1 is positioned the same (or worse) than non-winning application A2, then the applicant for A1 must pay at least p_{A2}.
- If A1 and C are both winning applications, together defeating application B that is in direct contention with both A1 and C, then each winning applicant must pay at least its proportionate share of the amount needed to defeat application B, i.e., A1 must pay at least $\left(\frac{p_{A1}}{p_{A1}+p_C}\right)p_B$ and C must pay at least $\left(\frac{p_C}{p_{A1}+p_C}\right)p_B$.
- More generally, if feasible set *S* of applications wins in lieu of feasible set *T* of applications, then each application X in set *S* must pay at least its proportionate share of the total amount needed to defeat set *T*, i.e. $\left(\frac{p_X}{\sum_{s\in S} p_s}\right)\sum_{t\in T} p_t$.
- If the hypotheses of more than one of the previous bullet points hold, then the conclusions of the respective bullet points are required to hold *in combination*. For example, if A1 is a winning application in the configuration depicted in Figure 3, then A1 is required to pay the *greater* of p_{A2} and $\left(\frac{p_{A1}}{p_{A1}+p_C}\right)p_B$.
- Each of these pricing conditions is applied in relation to the bids in the round in which the relevant non-winning application(s) is eliminated from the auction. For example, suppose in the configuration depicted in Figure 3 that application B is eliminated in Round 3 and that application A2 is eliminated in Round 5 of the auction. Then the requirement that C's payment is at least $\left(\frac{p_C}{p_{A1}+p_C}\right)p_B$ is based upon the bids in Round 3. Similarly, the requirement that A1's payment is at least $\left(\frac{p_{A1}}{p_{A1}+p_C}\right)p_B$ is based upon the bids in Round 3. However, the requirement that A1's payment is at least p_{A2} is based upon the bids in Round 5.

5. Information policy

In the ascending clock auction design for simple contention sets (and in most ascending-clock auctions generally), after each round, bidders are informed of the aggregate number of bidders remaining in the auction, but they are not provided with any disaggregated information such as the identities of the remaining bidders or the amounts of any individual bids (except to the extent that disaggregated information is revealed by the amount of the winners' payments). Since, in the contention sets with indirect contentions, many or most of the applications are positioned the same as each other, we will follow the same information policy generally, with one exception.

In some scenarios, an application may find itself with no remaining direct contentions while the overall auction is still continuing. For example, in the configuration depicted in Figure 3, if application B is eliminated before the end of the auction, then application C has no remaining direct contentions and will surely be selected as a winner. In such situations, the applicant for C will not be requested or permitted to place any further bids. By necessity, the auctioneer will need to disclose to the applicant for C the fact that it has no remaining direct contentions. And, out of concerns for treating bidders symmetrically, the auctioneer will also disclose this fact to all other bidders in the contention set. As such, all bidders in the contention set will know at the same time that the applicant for C is sure to win and will not be placing any further bids.

6. Examples

In this section, we provide three hypothetical examples of the recommended auction design for indirect contentions in operation.

Example 1. There are four bidders, configured as in Figure 3. Consider the following hypothetical round results:

Round	End-of-Round Price	A1	A2	B	C
1	$ 400,000	$ 400,000	$ 400,000	$ 400,000	$ 400,000
2	$ 900,000	$ 600,666	$ 700,000	$ 900,000	$ 500,000
3	$1,500,000	Eliminated	$ 751,111	$1,400,444	$ 552,222
Result		Eliminated	Eliminated	Winner: pays $1,303,333	Eliminated

Notes:

In Round 1, each application bid the end-of-round price. By condition (1) of Section 3, all four bidders remain in the auction.

In Round 2, application B bid the end-of-round price and, by condition (1) of Section 3, B remains in the auction. By condition (2) of Section 3, application A1 is eliminated from the auction, since it failed to meet the end-of-round price, while A2 is positioned the same as A1 but bid higher. By condition (3) of Section 3, applications A2 and C remain in the auction, as the sum of the bids for feasible set {A2, C} equaled $1,200,000, which is at least the end-of-round price of $900,000 for Round 2.

In Round 3, no application meets the requirements to remain in the auction for a further round: neither the feasible set {A2, C} ($1,303,333) nor the feasible set {B} ($1,400,444) has met the end-of-round price of $1,500,000 for Round 3. Hence, this is the final round of the auction. From the applications remaining in the auction after Round 2, the feasible set that maximizes the sum of the bids is determined to be {B}.

Application B is deemed to be the winner and is required to pay the amount of the highest losing feasible set, $p_{A2} + p_C$. Application B's payment is $1,303,333. There is no cost allocation issue, since B is the only winner.

Example 2. There are four bidders, configured as in Figure 3. Consider the following hypothetical round results:

Round	End-of-Round Price	A1	A2	B	C
1	$ 400,000	$ 400,000	$ 400,000	$ 400,000	$ 400,000
2	$ 900,000	$ 900,000	$ 900,000	$ 900,000	$ 500,000
3	$1,500,000	$1,500,000	$1,500,000	$1,400,000	$ 600,000
4	$2,200,000	$2,200,000	$2,200,000	Eliminated	Won
5	$3,000,000	$3,000,000	$2,600,666	Eliminated	Won
Result		Winner: pays $2,600,666	Eliminated	Eliminated	Winner: pays $1

Notes:

In Round 1, each application bid the end-of-round price. By condition (1) of Section 3, all four bidders remain in the auction.

In Round 2, the first three applications bid the end-of-round price and remain in the auction. By condition (3) of Section 3, the application for C also remains in the auction, as it is part of feasible sets {A1, C} and {A2, C}, each of which has the property that the sum of the bids of applications in the set equals at least the end-of-round price of Round 2.

In Round 3, applications A1, A2 and C remain in the auction, for the same reasons as in Round 2. However, application B fails condition (1) of Section 3, while satisfying condition (2), as applications A1 and A2 are positioned better than B yet bid higher. Hence, application B is eliminated from the auction. Whether A1 or A2 win the auction, they will be required to pay at least B's final bid.

In Round 4, there is no longer any application in direct contention with C. Therefore, C is not required or permitted to submit any more bids. Applications A1 and A2 bid the end-of-round price and remain in the auction.

In Round 5, application A1 bid the end-of-round price and remains in the auction. By condition (2) of Section 3, application A2 (which failed to meet the end-of-round price and is positioned the same as A1 but bid less) is eliminated.

The winning applications are A1 and C. Application A1 must pay the greater of $2,600,666 (as determined in Round 5) and $1,400,000 (as determined in Round 3). Application C must pay $1 (as determined in Round 3).

Example 3. There are 6 bidders, configured similarly to the SHOP / SHOPPING / 通販 contention set [5] and depicted in Figure 4:

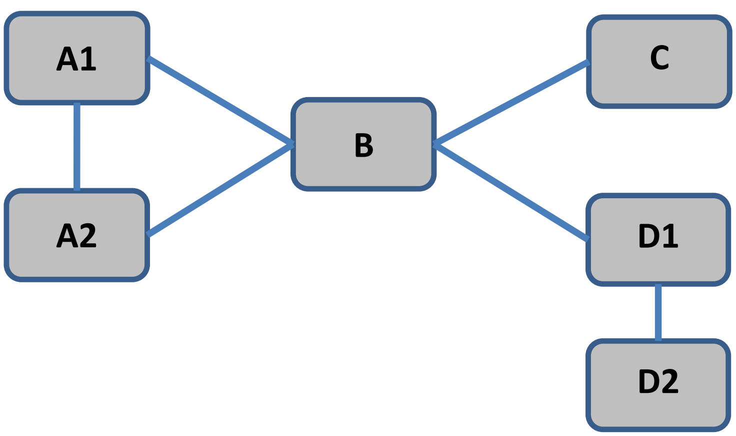

Figure 4

[5] Application B, the "cut vertex," has the position of Commercial Connect's application for .SHOP. Applications A1 and A2 have the positions of the other .SHOP applications. Application C has the position of the application for 通販. Application D1 has the position of Sea Tigers' application for .SHOPPING, while application D2 has the position of Uniregistry's application for .SHOPPING.

Consider the following hypothetical round results:

Round	End-of-Round Price	A1	A2	B	C	D1	D2
1	$ 400,000	$ 400,000	$ 400,000	$ 400,000	$ 200,000	$ 400,000	$ 400,000
2	$ 900,000	$ 900,000	$ 750,111	$ 900,000	$ 200,000	$ 900,000	$ 900,000
3	$1,500,000	$1,200,000	Eliminated	$1,500,000	$ 800,000	$1,500,000	$1,500,000
4	$2,200,000	$1,500,000	Eliminated	$2,200,000	$1,100,000	$1,700,777	$1,900,000
5	$3,000,000	$1,500,000	Eliminated	$2,666,000	$1,500,000	Eliminated	Won
Result		Winner: pays $1,333,000	Eliminated	Eliminated	Winner: pays $1,333,000	Eliminated	Winner: pays $1,700,777

Notes:

In Round 1, each of the five applications other than C bid the end-of-round price. By condition (1) of Section 3, these five bidders remain in the auction. By condition (3) of Section 3, the application for C also remains in the auction, as it is an element of feasible sets {A1, C, D1}, {A2, C, D1}, {A1, C, D2} and {A2, C, D2}—for any of these, the sum of the bids is at least the end-of-round price of Round 1.

In Round 2, applications A1, B, D1 and D2 bid the end-of-round price and remain in the auction. By condition (3) of Section 3, the application for C also remains in the auction, by the same reasoning as after Round 1. However, by condition (2) of Section 3, application A2 (which failed to meet the end-of-round price, but is positioned the same as A1 and bid less) is eliminated. On this basis, A1's payment (if A1 wins) is required to be at least $750,111.

In Round 3, applications B, D1 and D2 bid the end-of-round price and remain in the auction. By condition (3) of Section 3, the applications for A1 and C also remain in the auction, as they are elements of feasible sets {A1, C, D1} and {A1, C, D2}—for either of these, the sum of the bids ($3,500,000) equals at least the end-of-round price of Round 3 ($1,500,000).

In Round 4, application B bid the end-of-round price and remains in the auction. The applications for A1 and C also remain in the auction, by similar reasoning as in Round 3. However, by condition (2) of Section 3, application D1 (which failed to meet the end-of-round price, but is positioned worse than D2 and bid less) is eliminated. Application D2 is a winner, as there is no longer any application in direct contention with D2. The payment of D2 is $1,700,777, the final price of D1.

In Round 5, the winner D2 is not required or permitted to submit any more bids. None of the bidders met the end-of-round price individually. The applications for A1 and C remain in the auction, as they are parts of feasible set {A1, C }, which has the property that the sum of the bids of applications in the set ($3,000,000) equals at least the end-of-round price of Round 5 ($3,000,000). [We do not consider the set {A1, C, D2}, since D2 has already been won in the previous round.] Application B does not remain in the auction, as B is not part of any feasible set *S* of applications that remained in the auction in Round 4 for which the sum of the bids equals at least the end-of-round price of Round 5. [Again, we do not consider the set {B, D2}, since D2 has already been won.]

Hence, this is the final round of the auction. From the applications that remained in the auction and had not won after Round 4, the feasible set that maximizes the sum of the bids is determined to be {A1, C}. (This set sums to $3,000,000; whereas the only alternative, {B}, sums only to $2,666,000.)

Applications A1 and C must pay at least their proportionate shares of the amount needed to defeat B. Note that A1 and C together bid $3,000,000, but only $2,666,000 was needed to meet B's bid. Hence, each pays at least $\left(\frac{1{,}500{,}000}{1{,}500{,}000+1{,}500{,}000}\right)2{,}666{,}000 = 1{,}333{,}000$.

We conclude that A1 must pay the greater of $1,333,000 (as determined in Round 5) and $750,111 (as determined in Round 2). Meanwhile, Application C pays $1,333,000 (as determined in Round 5). Recall that application D2 was also determined in Round 4 to be a winner and was required to pay $1,700,777, the final bid by application D1.

7. Conclusion

This paper has described an ascending clock auction design for resolving string contention among competing applications for new gTLDs in contention sets where indirect contentions are present. It is recommended because it has several important strengths:

- **Efficiency.** In simple contention sets, the economic rationale for the auction was that it yields efficient outcomes, in the sense of allocating the string to the applicant with the highest value. In contention sets with indirect contentions, the auction may also be determining which strings are allocated. In determining the allocation, the same rationale implies that the auction should attempt to maximize the sum of the winning applicants' values. For example, in the "A—B—C" scenario of Figure 2, the auction is determining whether the applicant for B should alone be satisfied, or whether the applicants for A and C should both be satisfied. This depends on the comparison between B's value, and the sum of A's and C's value. For example, consider the values in Table 1:

Applicant	Value
A	$2 million
B	$3 million
C	$2 million

Table 1

With the values in Table 1, efficiency would favor satisfying applicants A and C, since 2 + 2 > 3. Conversely, consider the values in Table 2:

Applicant	Value
A	$2 million
B	$7 million
C	$2 million

Table 2

With the values in Table 2, efficiency would favor satisfying applicant B alone, since 7 > 2 + 2.

The recommended auction design systematically tests which feasible set of applications attains the highest sum of values, so the design will make the efficient allocation with high probability. Alternative designs might merely seek to identify the application with the single highest value—with data as in Table 1, this would select applicant B as the sole winner, the wrong choice in relation to efficiency. Other alternative designs might merely favor awarding the maximum number of strings—with data as in Table 2, this would tend to select applicants A and C as winners, the wrong choice in relation to efficiency.

- **Simplicity of bidding.** A key advantage of the ascending clock auction design for simple contention sets—those with only direct contentions—was the simplicity of bidding. Each round, the bidder only needed to decide whether its value exceeded the end-of-round price; if it did, the bidder would bid the end-of-round price; while if it did not, the bidder would bid its value.

The proposed auction design for indirect contentions—while somewhat more complicated—is still relatively simple for bidders compared to alternative designs. Participation will be especially easy for bidders who are already familiar with ascending clock auctions, from participating in prior auctions for simple contention sets.

- **Consistency with the Applicant Guidebook (AGB).** The ascending-clock auction recommended in the present paper has a variety of advantages, so that it stands on its own merits. However, it should also be noted that Section 4.3.1 of the AGB specifies:

  > An auction of two or more applications within a contention set is conducted as follows. The auctioneer successively increases the prices associated with applications within the contention set, and the respective applicants indicate their willingness to pay these prices. As the prices rise, applicants will successively choose to exit from the auction. When a sufficient number of applications have been eliminated so that no direct contentions remain (i.e., the remaining applications are no longer in contention with one another and all the relevant strings can be delegated as TLDs), the auction will be deemed to conclude. At the auction's conclusion, the applicants with remaining applications will pay the resulting prices and proceed toward delegation. This procedure is referred to as an "ascending-clock auction."[6]

  The current recommendation adheres closely both to the letter and the spirit of the AGB. By contrast, any mechanism for resolving string contention other than an ascending-clock auction would introduce the risk that one or more applicants might raise objections to the mechanism as contrary to the AGB. This would potentially introduce further delays into the process for allocating new gTLDs that has already taken many years.

- **Capability of Proxy Bidding.** This ascending clock auction design for complex contention sets will be implemented with the same proxy bidding capabilities as in the ascending clock auctions for simple contention sets. For all purposes of the auction, including the calculation of payments, a proxy bid greater than the end-of-round price will continue to be treated the same as a regular (non-proxy) bid equaling the end-of-round price. As such, the auction implementation will continue to allow bidders the meaningful choice between real-time bidding in a dynamic auction and submitting one's bids at the very beginning, as if it were a sealed-bid auction. Use of a proxy bid will not be held against the bidder.

- **No "Revival".** The AGB specifies in Section 4.3.1 that "[i]f a bidder exited the auction in a previous auction round, the bidder is not permitted to re-enter in the current auction round," and the current recommendation adheres to this requirement. In the case of indirect contention, it might be argued that—using the A-B-C example of Figure 2, above—an applicant C may fail to meet the end-of-round price and thus be eliminated, but could later also be "revived" if its indirect counterpart, application A, prevails in the auction, because applications A and C may co-exist. The concept of a feasible set and condition (3) of Section 3 prevents such revival. Application C is deemed to have exited the auction only when it, together with application A, has failed to meet the end-of-round price and application B has submitted a higher bid than A and C together. At that point, application B prevails. In the opposite case, applications A and C may still have not met the end-of-round price but remain in the auction together as a feasible set whose bids

[6] ICANN, *gTLD Applicant Guidebook* (version 2012-06-04), module 4, p. 4-20.

together would be the highest bid. Their applications would then prevail over application B, and their strings would both delegated. In neither case does an application exit and then later re-enter.